



**Barcelona
Supercomputing
Center**

Centro Nacional de Supercomputación

PoTrA: Power Trace Analyzer

A Framework for Building Power Models For
Next Generation Multicore Architectures

Outline

« Introduction	(10 min.)
« Data Gathering	(25 min.)
« Analysis	(45 min.)
« Modeling	(45 min.)
« Case of Use	(45 min.)

PoTrA: research-oriented tool

- ⌘ Data processing and visualization tool, but with specific support for HW counters
 - Developed by Ramon Bertran (and Lluís Vilanova, Victor Jimenez)
 - PhD students @BSC
- ⌘ Collect multiple data sources (**CSV files**)
- ⌘ Generate an N-dimensional data object: potradata
 - Each dimension matches a “**configuration parameter**”
- ⌘ Different usages
 - Data transformation (**processing**)
 - Data visualization (**plots**)
- ⌘ Behavior is defined through **configuration files**
 - **Text readable files**

PoTrA: research-oriented tool

« Example:

- Collect samples of HW counters for each application in the SPECOMP
 - for each F in [0,8 1,2 1,6 2,0 2,5]
 - for each B in SPECOMP
 - » for each P in [1 2 4 6 8]
 - Run B with P cores, collect **300 samples** of HW counters -> around 500 .csv files
- Process data
 - Example: Normalize to (0.8GHz x 1 core)
- Visualize Data

Other options

« Excel / LibreOffice

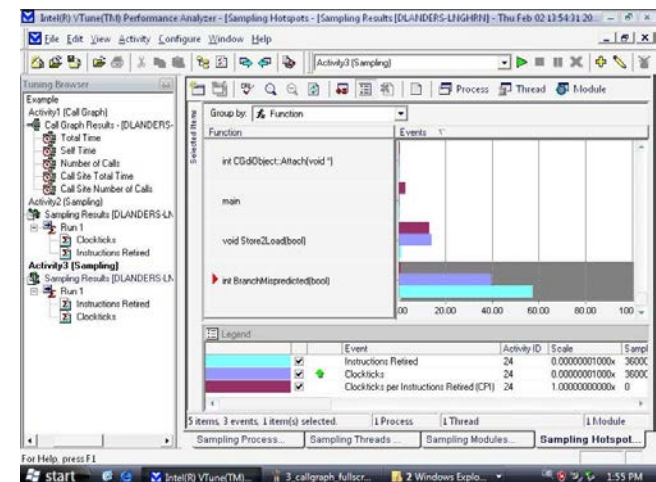
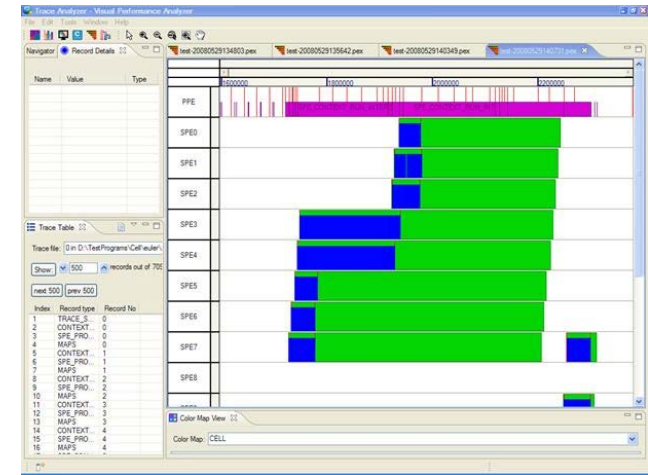
- Do not scale (they are okay for small datasets)
- Copy & paste is error prone

« PAPI, Oprofile, ...

- low level, main support for configuring and reading the counters

« Intel Vtune, IBM PDT VPA

- high level, gather PMCs, built-in metrics and analyses. Mainly target performance optimization.



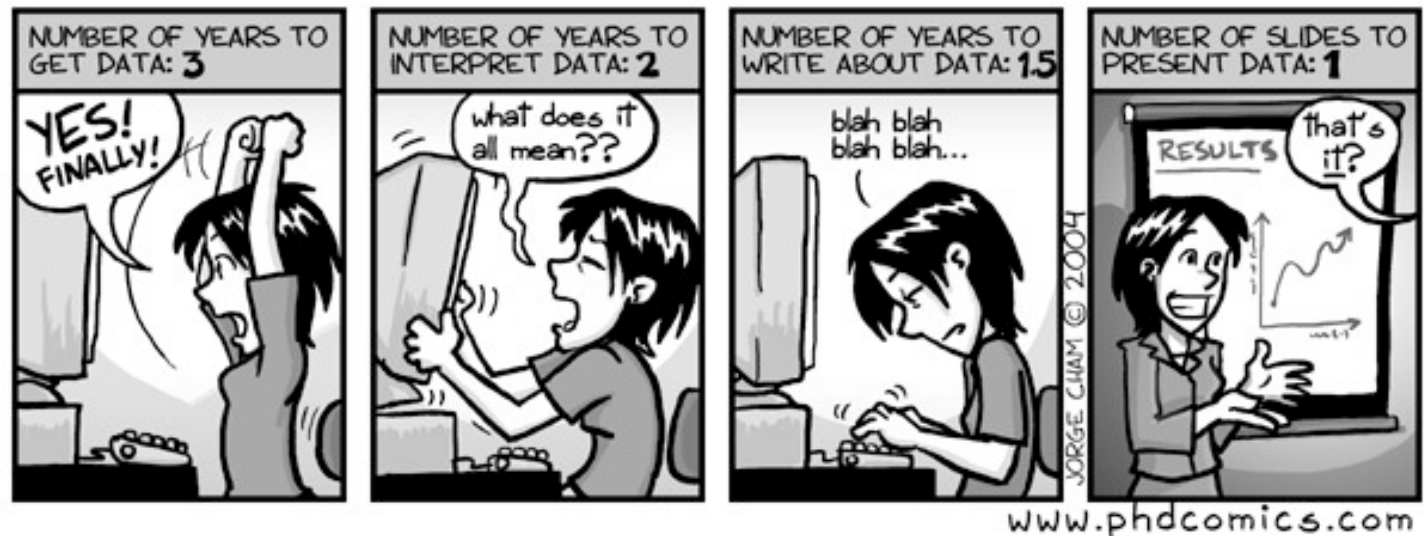
Why PoTrA is different and worth ?

« Specifically designed to operate with HW counters

« Research oriented

- Support for PhD

DATA: BY THE NUMBERS



What can be done with counters ?

« Create a detailed description of what is happening in the processor

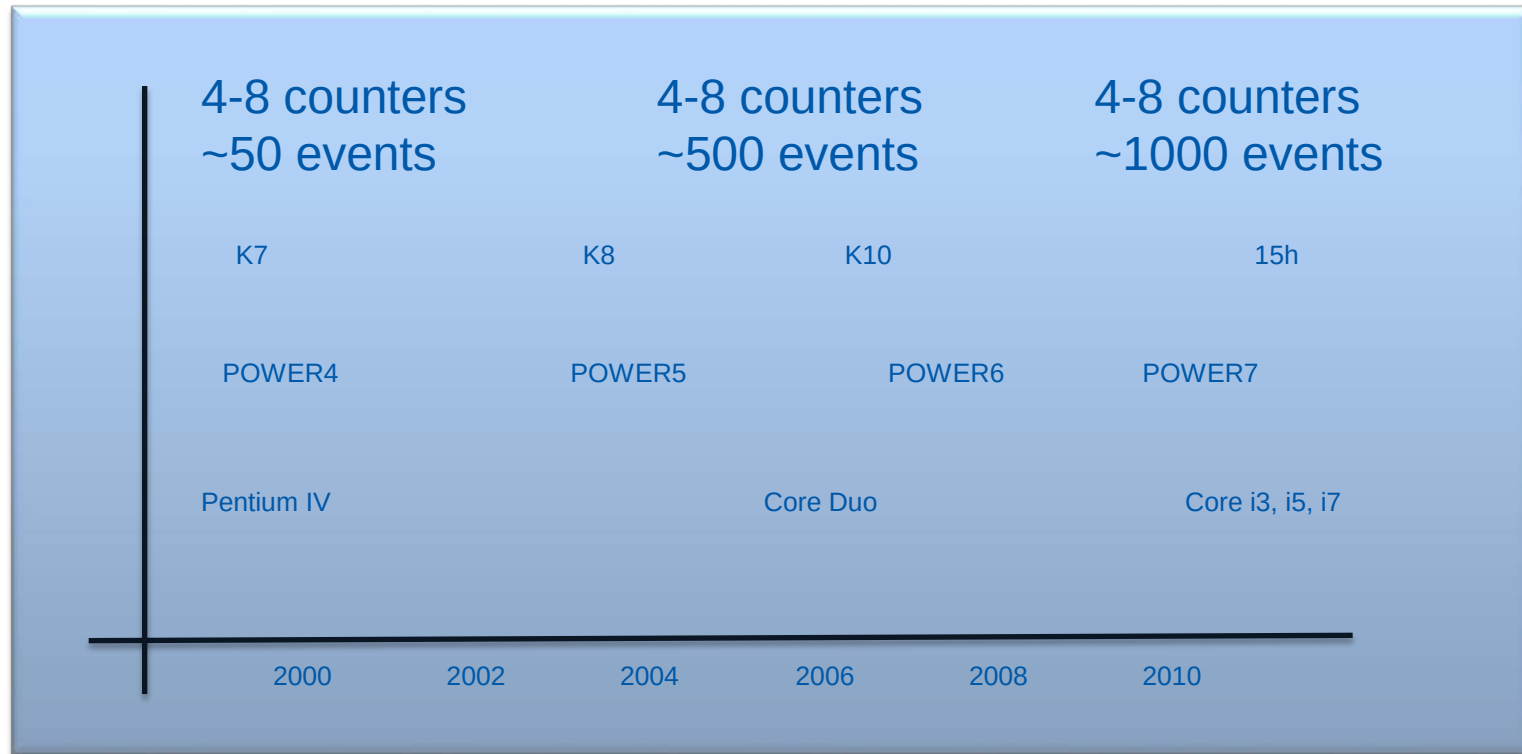
- Black-box
- Hooks for peeking into micro-architectural activities
- Available separately for each core
- Fairly ubiquitous
- Easily accessible

« Direct application for

- **Power and energy**
 - Modeling and accounting
- Performance optimization
 - CPI stack

Evolution of Hardware Counters and Events

- « Low number of counters: 4-8 grouped
- « Significant number of events



References for the tutorial

- “ F. Bellosa. “The benefits of event: driven energy accounting in power-sensitive systems”, In EW '00, Kolding, Denmark, 2000.
- “ C. Isci et al. “Runtime power monitoring in high-end processors: Methodology and empirical data”, In MICRO'03, San Diego, CA, USA, 2003.
- “ R. Bertran et al. “Decomposable and responsive power models for multicore processors using performance counters”, In ICS'10, Tsukuba, Ibaraki, Japan, June 2010
- “ R. Bertran et al. “Energy accounting for shared virtualized environments under DVFS using pmc-based power models”, Future Generation Computer Systems, pp457-468, 2012

Outline

« Introduction	(10 min.)
« Data Gathering	(25 min.)
« Analysis	(45 min.)
« Modeling	(45 min.)
« Case of Use	(45 min.)

DATA GATHERING

- « Meters
- « Platform setup
- « Hardware counters and events
- « Tool: pmctrace

Meters

Parameters

- Min, Max
- Inertia, threshold

Power meter :

- Pros: non-intrusive, off-the-shelf, easily accessible by software
- Cons: no isolation of CPU power, lower sampling rate, lower sensitivity to power changes

Sensors

- Pros: higher granularity, potentially higher sampling rates
- Cons: custom hardware, relatively complex software interfaces

Meters: WattsUp

- ⌘ Max sampling rate: 1 sample/sec
- ⌘ Accuracy: +/- 1.5%
- ⌘ Linux interface: wattsup-daemon
- ⌘ Talks to meter over USB



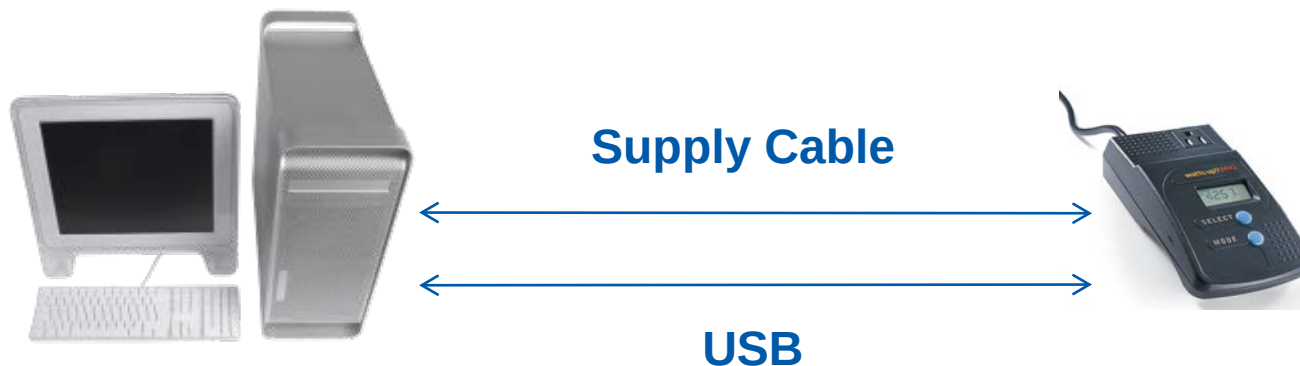
Platform Set-up: Connecting the meter

Target Machine

- Runs μ benchmarks and test benchmarks
- Collects counter data
- Runs a limited OS image
 - no network
 - no display (in case a laptop)
 - all daemons are stopped

Test Machine

- Runs meter-daemon
- Collects samples from power meter



Platform Set-up: Connecting the meter

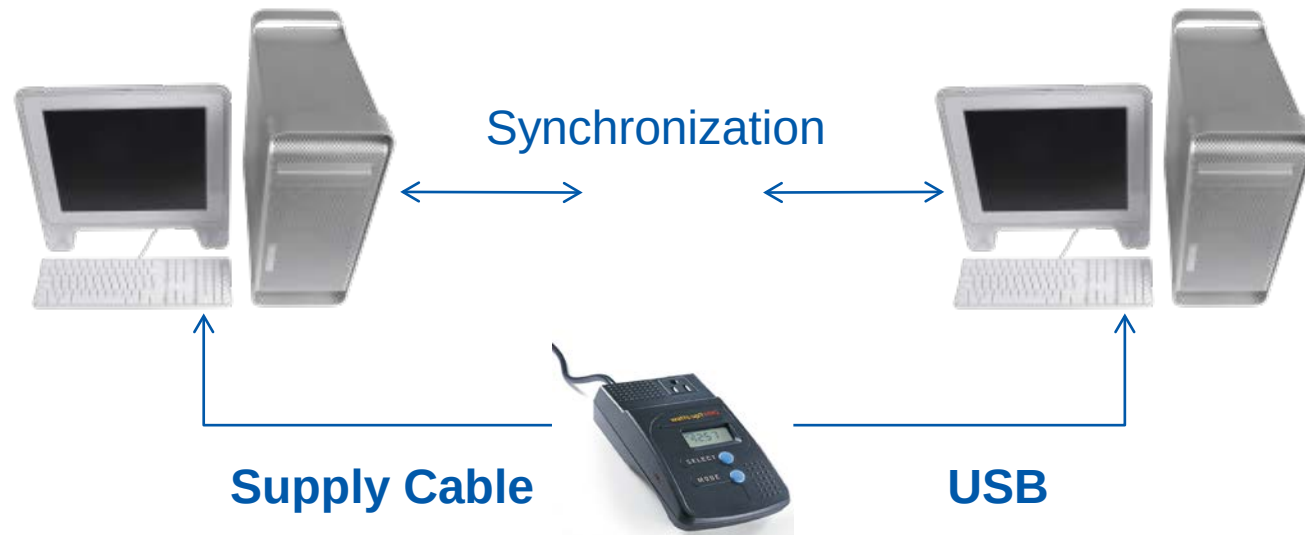
Target Machine

- Runs μ benchmarks and test benchmarks
- Collects counter data
- Runs a limited OS image
 - no network
 - no display (in case a laptop)
 - all daemons are stopped

≠

Test Machine

- Runs meter-daemon
- Collects samples from power meter
- Runs a "comfortable" OS image



Platform Set-up: Linux OS image

« Kernel version

- >> 2.3.31, otherwise a patch is required

« Libraries

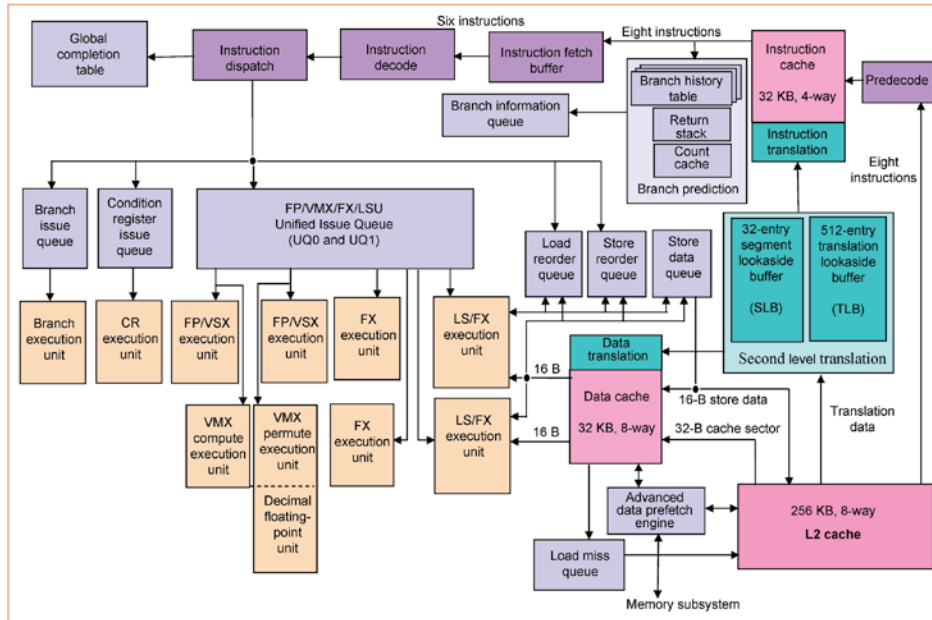
- libpfm3.x
- Add pfmon if linux < 2.6.30

« Virtualization is supported

- HW counters are associated to processes (safe for context switch)
- Both VMs and their processes can be traced from the HOST OS.

HW counters

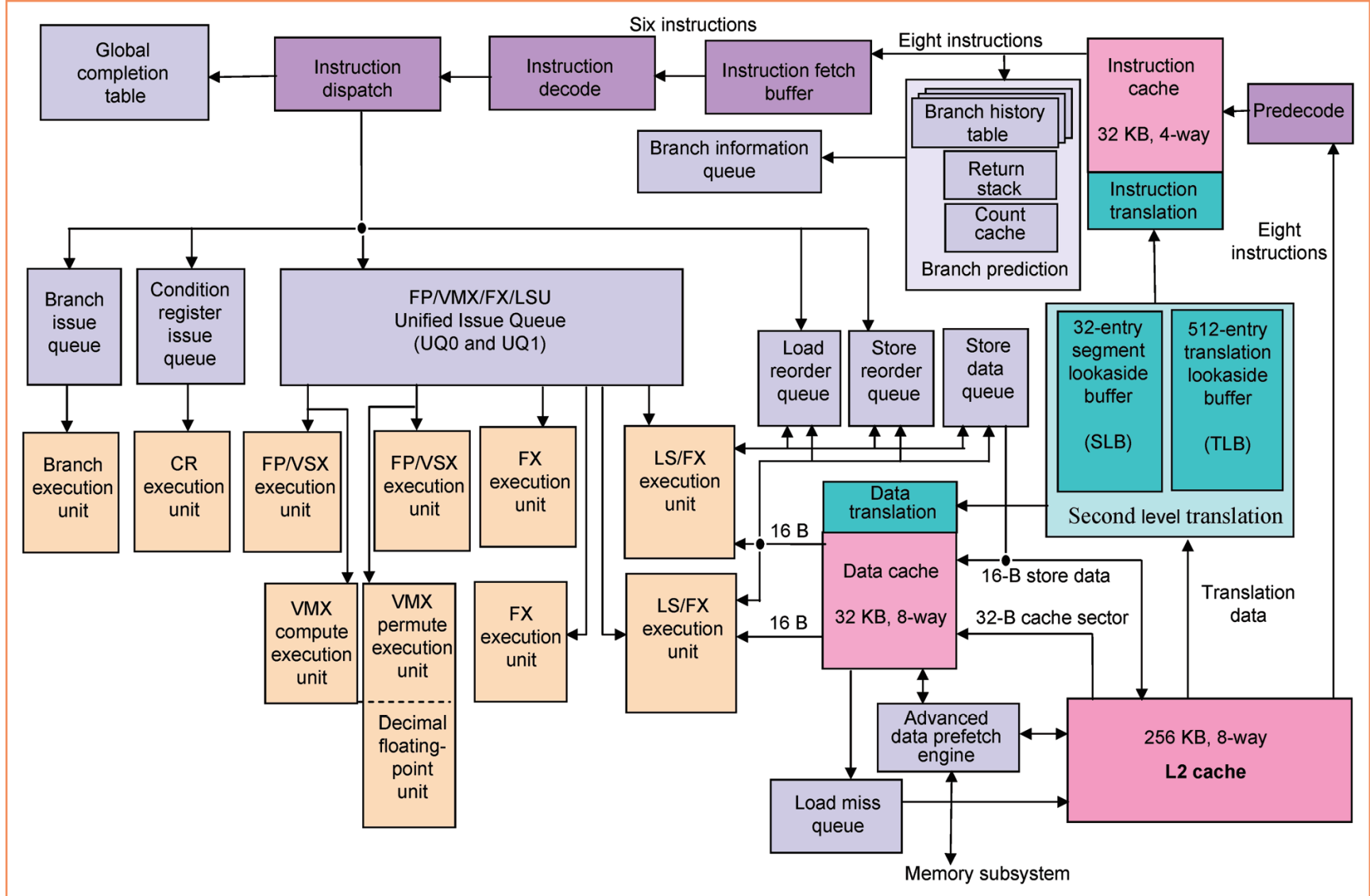
- ❧ To identify and understand the HW counters, follow the stages for executing one instruction
 - FETCH, DECODE, DISPATCH, ISSUE, EXECUTE, COMMIT
- ❧ Identify the blocks that are active in each stage
 - HW counters are associated to these blocks



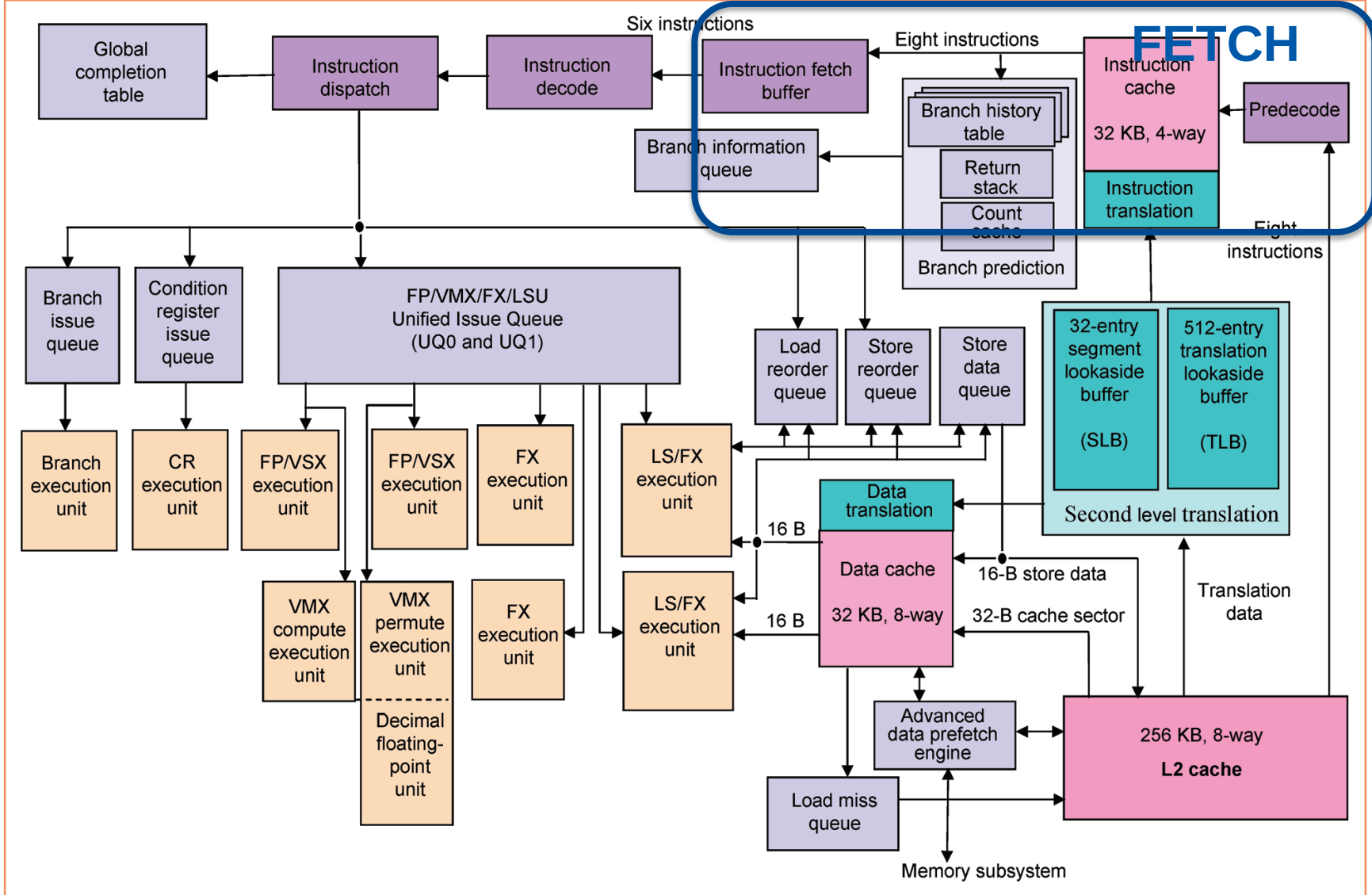
B. Sinharoy, et al. "IBM POWER7 multicore server processor"

IBM Journal of Research and Development, Volume: 55 , Issue: 3, Page(s): 1-29

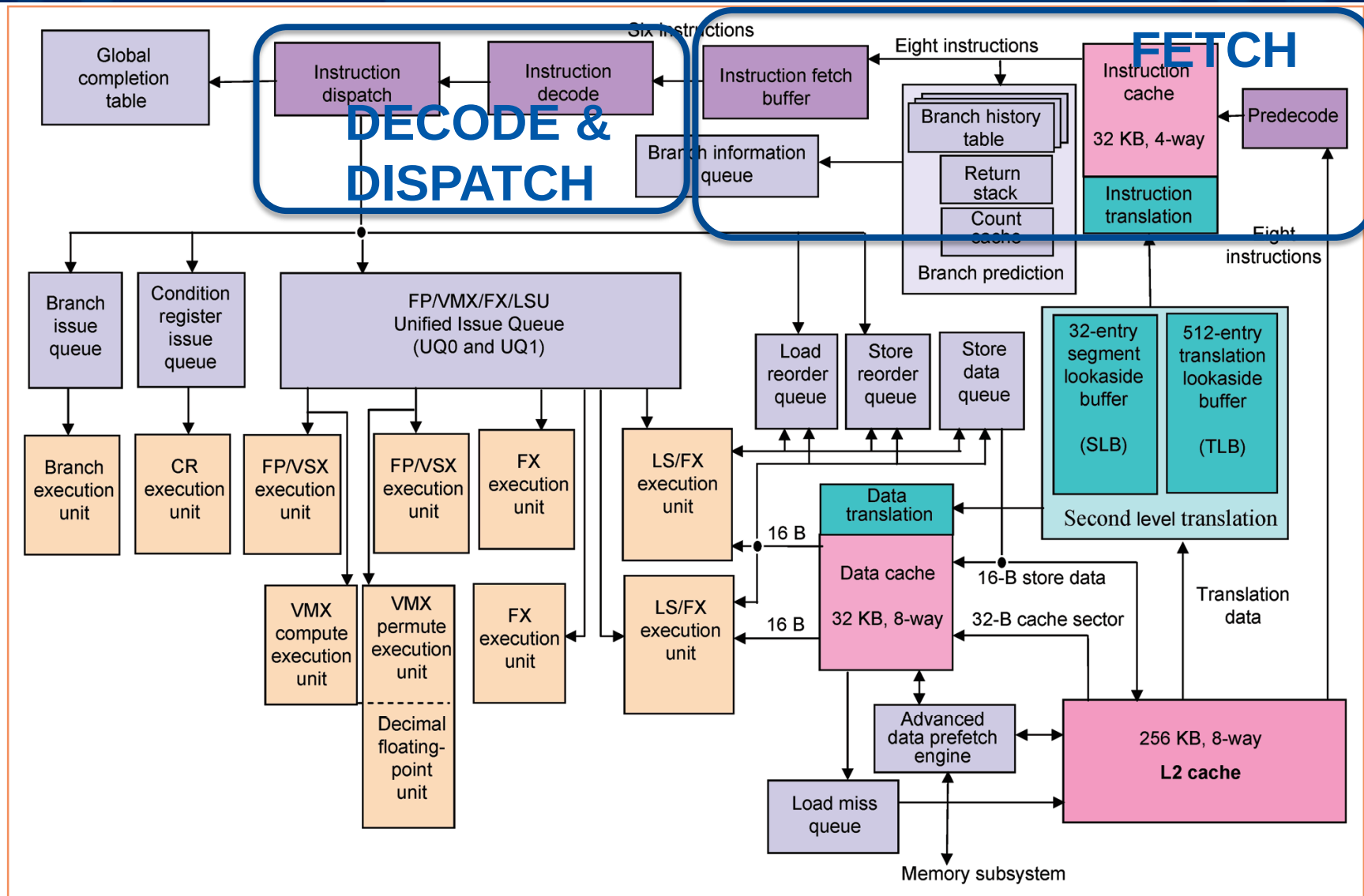
HW counters: IBM POWER7



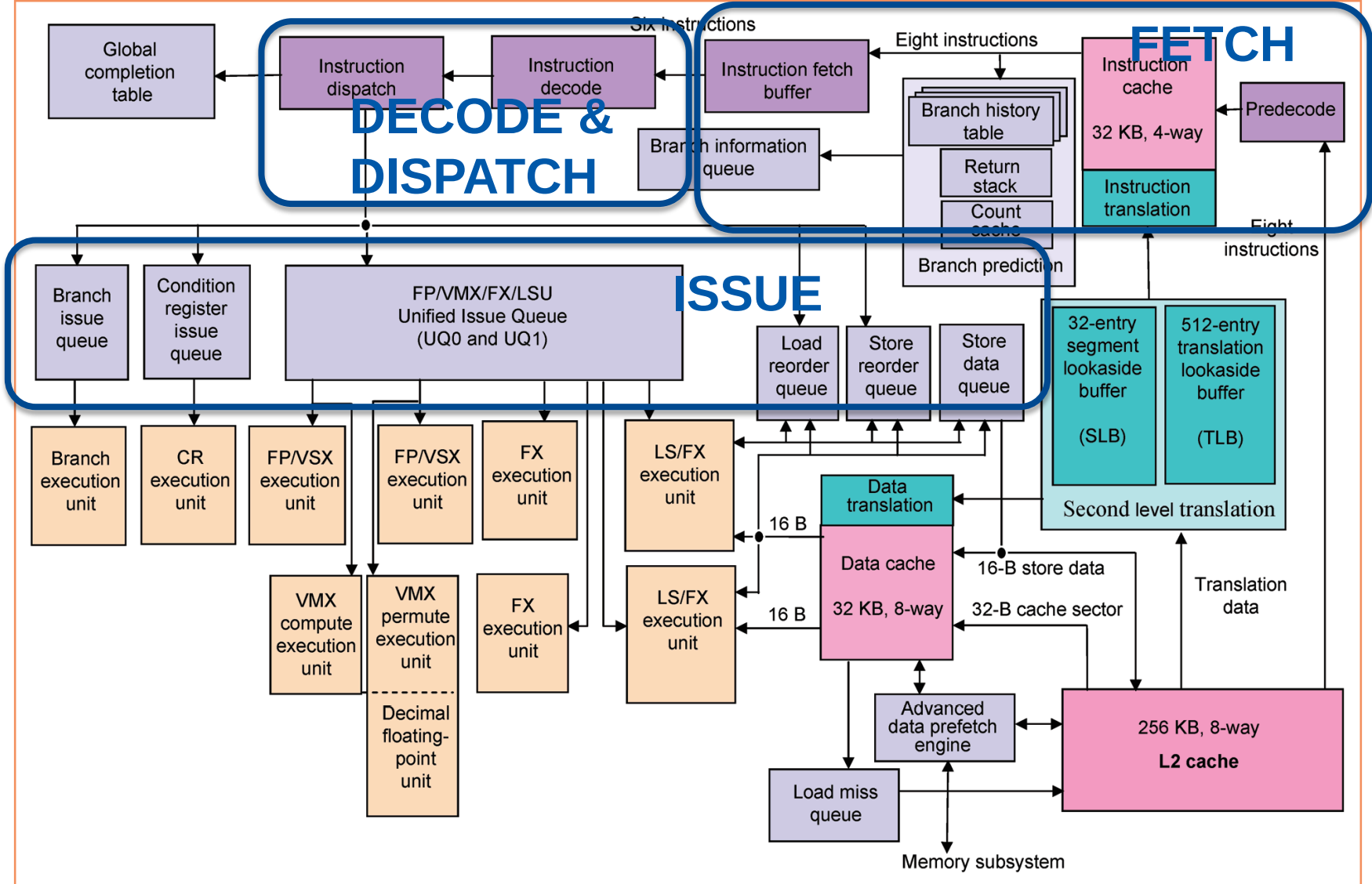
HW counters: IBM POWER7



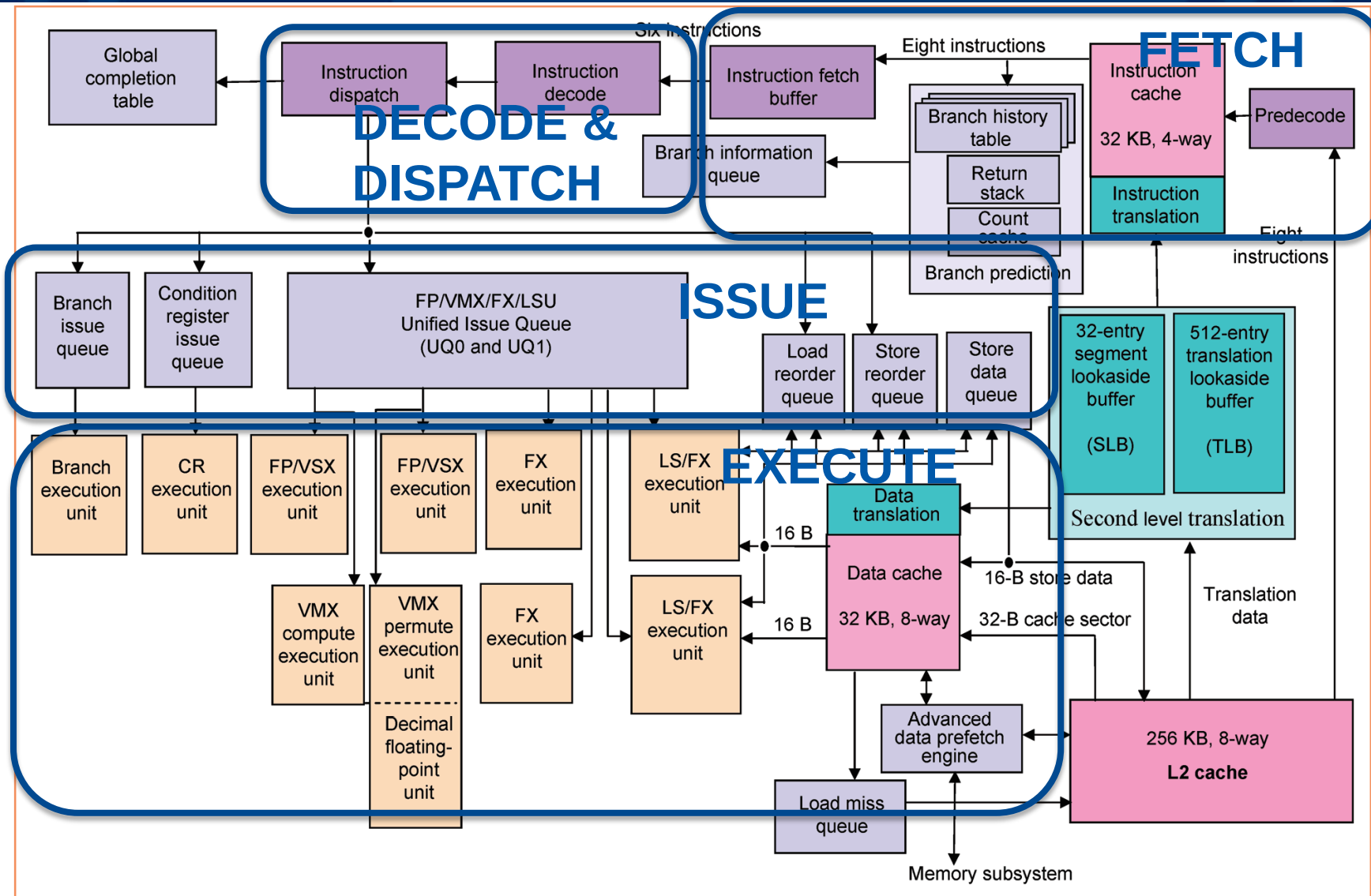
HW counters: IBM POWER7



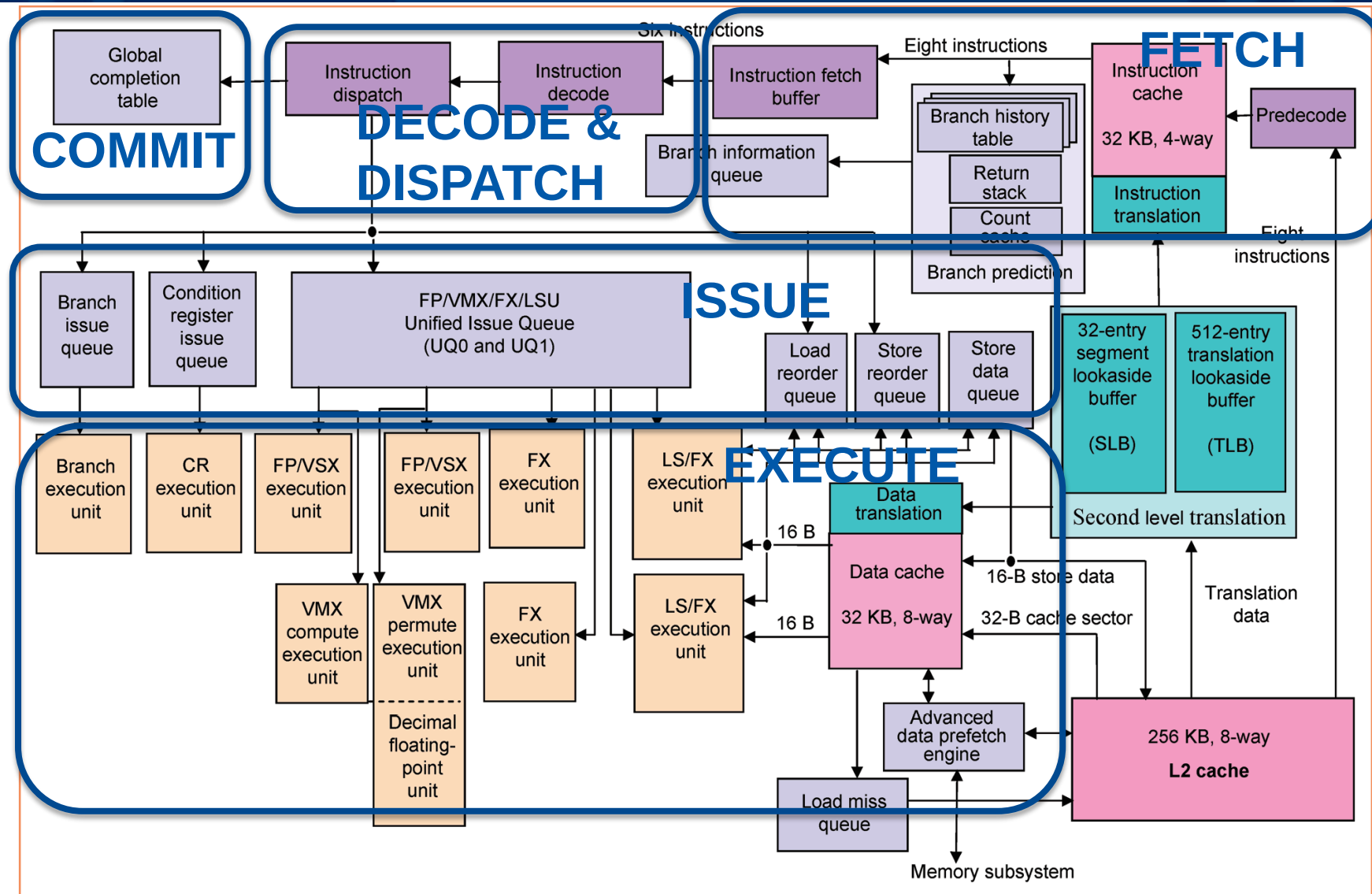
HW counters: IBM POWER7



HW counters: IBM POWER7



HW counters: IBM POWER7



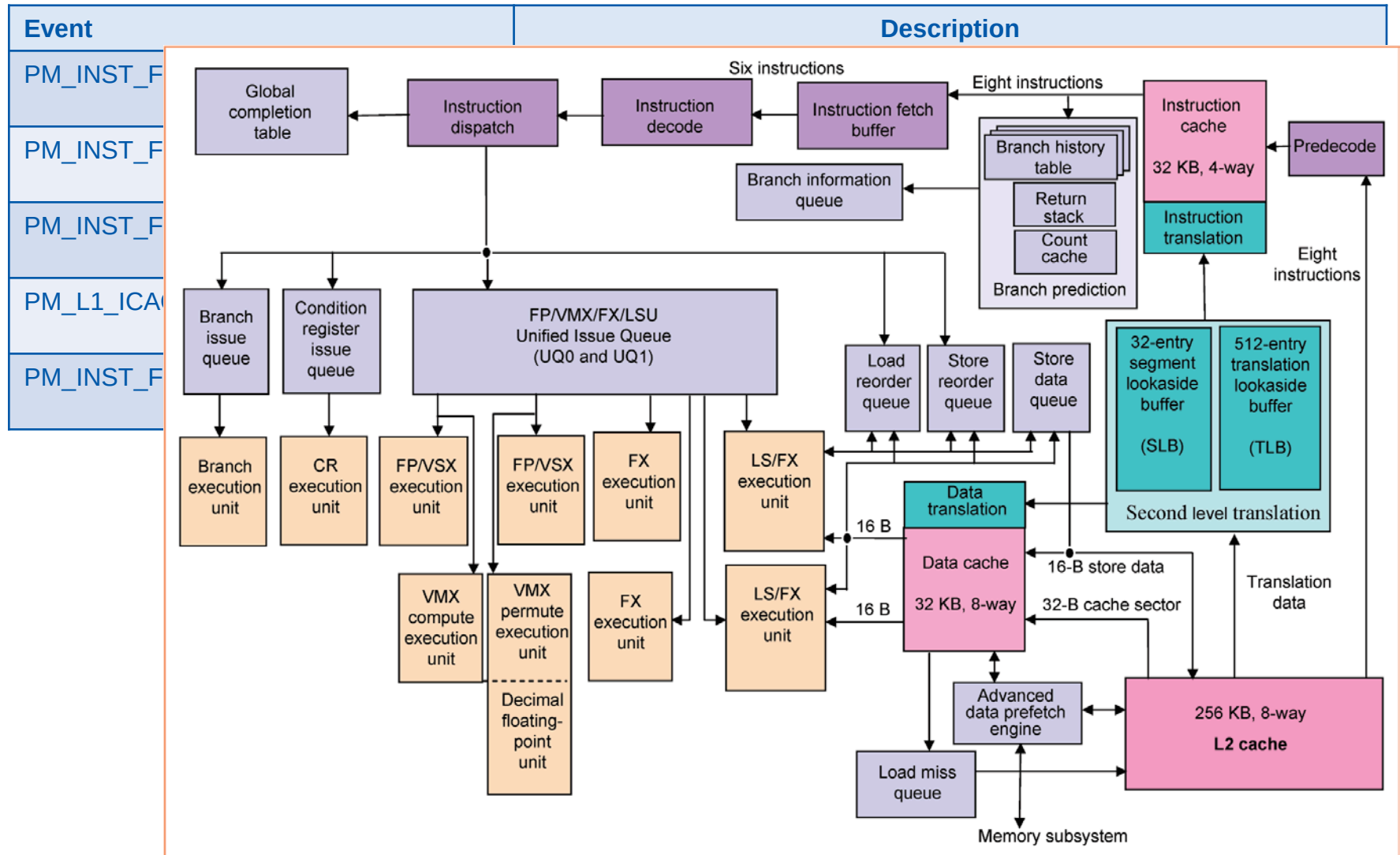
Pipeline & Instructions

fetch, decode, dispatch, issue, execute, commit

Event	Description
PM_INST_FROM_L2MISS	The processor's Instruction Cache was reloaded with data from the local chiplet's L2 cache due to a demand load.
PM_INST_FROM_L3MISS	The processor's Instruction Cache was reloaded with data from the local chiplet's L3 cache due to a demand load.
PM_INST_FROM_PREF	The processor's Instruction Cache was reloaded but not from the local L2 due to a demand load.
PM_L1_ICACHE_MISS	An instruction fetch group was fetched from L1. Fetch Groups can contain up to 8 instructions
PM_INST_FROM_DMEM	The processor's Instruction Cache was reloaded with data from a chip's memory on a different Node (Distant) due to a demand load.

Pipeline & Instructions

fetch, decode, dispatch, issue, execute, commit



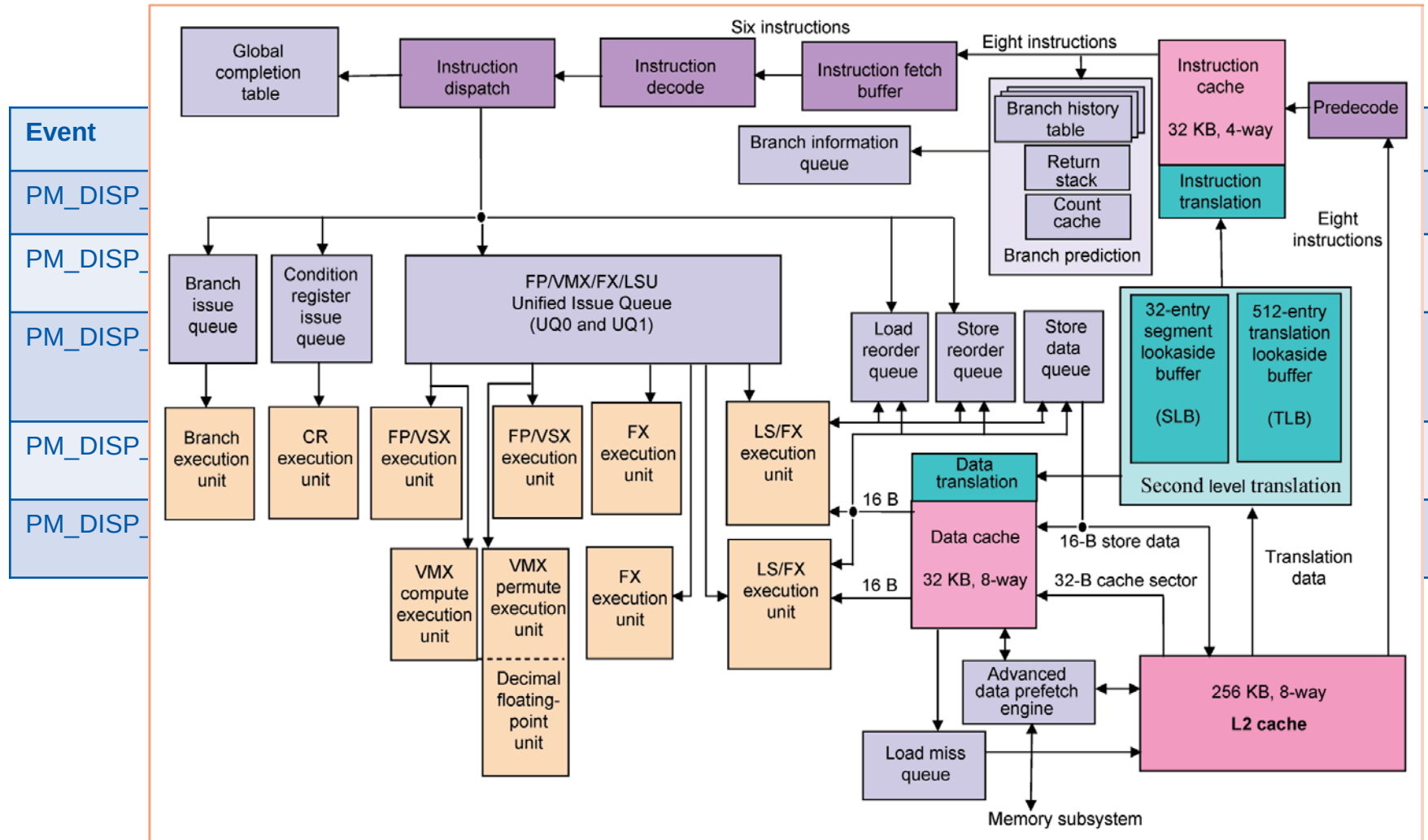
Pipeline & Instructions

fetch, decode, **dispatch**, issue, execute, commit

Event	Description
PM_DISP_HELD	Dispatch Held Cycles dispatch to pipeline was held
PM_DISP_WT	Dispatched Starved (not held, nothing to dispatch). Cycles dispatch to pipeline was not held and there were no groups to dispatch
PM_DISP_CLB_HELD_RES	An instruction group is held at dispatch in the ibuffer, due to any resource full condition including BRQ Full, CR Issue Q full, Unified Issue Q Full, Mapper full, LRQ Full, STQ full, or other resource holds such as syncs tlbie
PM_DISP_HELD_THERMAL	Dispatch Held due to Thermal Cycles dispatch to pipeline was held due to a thermal condition
PM_DISP_CLB_HELD_SB	Dispatch/CLB Hold: Scoreboard An instruction group is held at dispatch in the ibuffer, due to scoreboard full

Pipeline & Instructions

fetch, decode, **dispatch**, issue, execute, commit



Pipeline & Instructions

fetch, decode, dispatch, issue, **execute**, commit

« Branches

Event	Description
PM_BR_MPRED	Number of Branch Mispredicts. A branch instruction was predicted. This could have been a target prediction a condition prediction or both
PM_BR_MPRED_CR	Branch mispredict - taken/not taken. A conditional branch instruction was incorrectly predicted as taken or not taken. The branch execution unit detects a branch mispredict because the CR value is opposite of the predicted value. This will result in a branch redirect flush if not overfidden by a flush of an older instruction. The processor's Instruction Cache was reloaded but not from the local L2 due to a demand load.
PM_BR_MPRED_LSTACK	Branch Mispredict due to Link Stack. A branch instruction target was incorrectly predicted due to Link stack Misprediction. This will result in a branch mispredict flush unless a flush is detected from an older instruction.
PM_BR_PRED	Branch Predictions made. A branch prediction was made, counted at branch execute time

« Branches



29

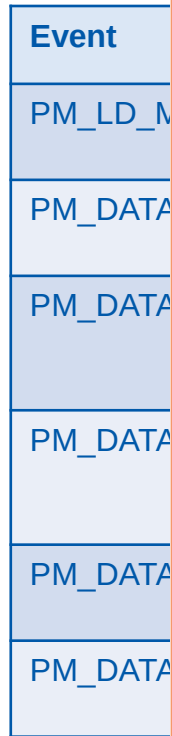
Pipeline & Instructions

fetch, decode, dispatch, issue, **execute**, commit

Memory Operations: LD/ST

Event	Description
PM_LD_MISS_L1	Load Missed L1. Load references that miss the Level 1 Data cache.
PM_DATA_FROM_L2	Data loaded from L2. The processor's Data Cache was reloaded with data from the local chiplet's L2 cache due to a demand load.
PM_DATA_FROM_DL2L3_MOD	Data loaded from distant L2 or L3 modified. The processor's Data Cache was reloaded with Modified (M) data from a chip's L2 or L3 on a different Node (Distant) due to a demand load .
PM_DATA_FROM_L21_SHR	Data loaded from another L2 on same chip shared. The processor's Data Cache was reloaded with Shared (S) data from another L2 on the same chip due to a demand load .
PM_DATA_FROM_LMEM	Data loaded from local memory. The processor's Data Cache was reloaded with data from the local chip's memory due to a demand load.
PM_DATA_FROM_DMEM	Data loaded from distant memory. The processor's Data Cache was reloaded with data from a chip's memory on a different Node (Distant) due to a demand load .

Memory Operations: LD/ST



Pipeline & Instructions

fetch, decode, dispatch, issue, **execute**, commit

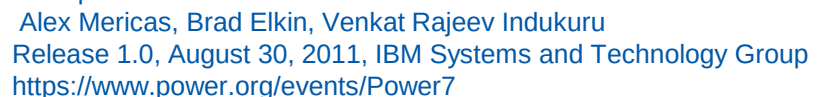
Memory Operations: LD/ST

- Address Translation

Event	Description
PM_DERAT_MISS_4K	DERAT misses for 4K page. The source page size of a DERAT reload is 4K
PM_DTLB_MISS	TLB reload valid. Data TLB misses, all page sizes.

Memory Operations: LD/ST

PM_DTLB



Pipeline & Instructions

fetch, decode, dispatch, issue, **execute**, commit

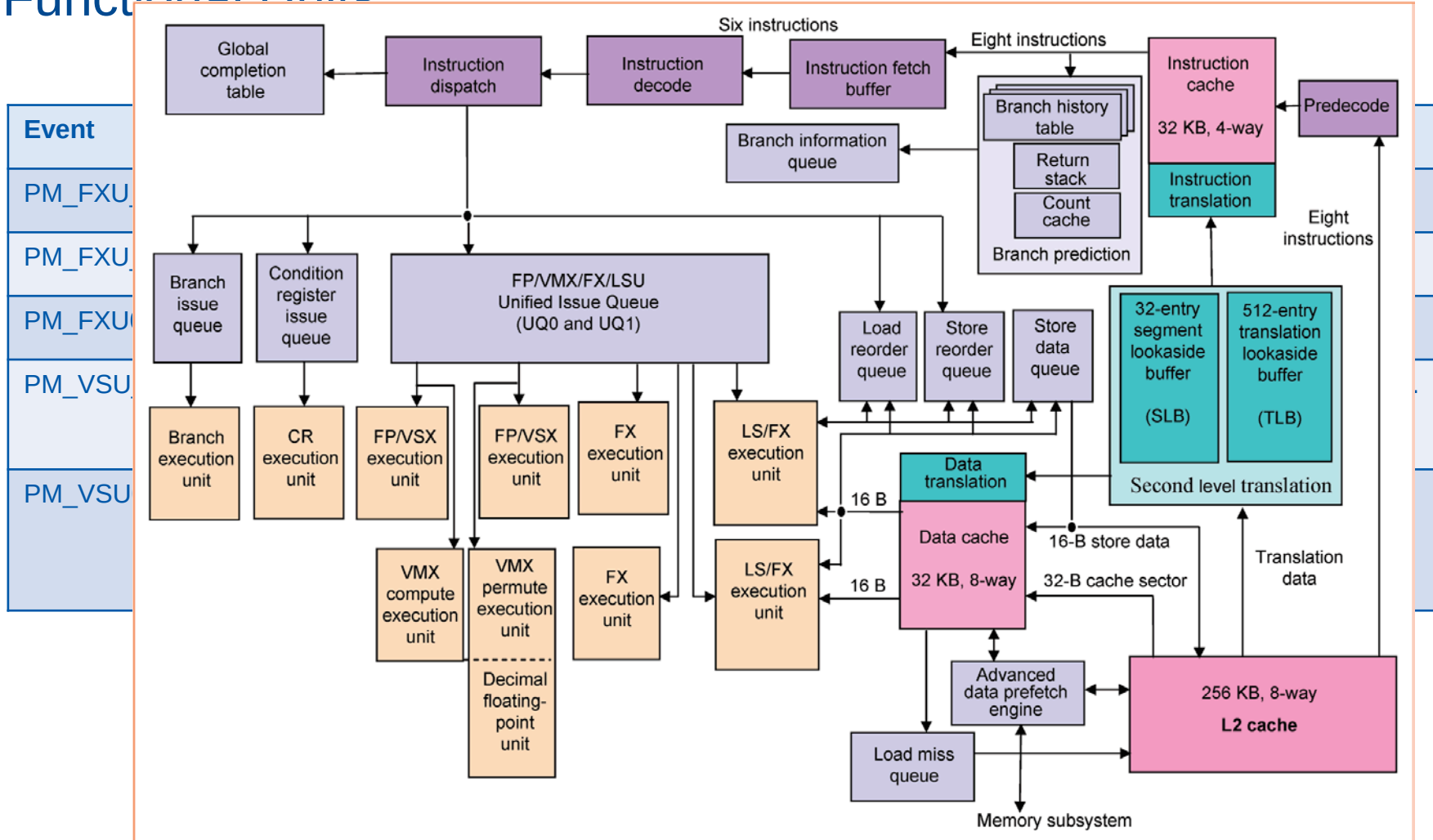
Functional Units

Event	Description
PM_FXU_BUSY	fxu0 busy and fxu1 busy. Cycles when both fixed point units are both busy
PM_FXU_IDLE	fxu0 idle and fxu1 idle FXU0 and FXU1 are both idle
PM_FXU0_BUSY_FXU1_IDLE	fxu0 busy and fxu1 idle Cycles when FXU0 busy and FXU1 idle
PM_VSU_1FLOP	One flop (fadd, fmul, fsub, fcmp, fsel, fabs, nabs, fres, fsqrte, fneg) operation finished. The vector scalar has finished one flop (fadd, fmul, fsub, fcmp, fsel, fabs, fnabs, fres, fsqrte, fneg, xsadd, xsmul, xssub, xscmp, xssel, xsabs, xsnabs, xsre, xssqrte, xsneg)
PM_VSU0_8FLOP	Eight flops operation (DP vector versions of fdiv,fsqrt and SP vector versions of fmadd,fnmadd,fmsub,fnmsub). The vector scalar unit 0 (VSU0) has finished eight flops operation (DP vector versions of fdiv,fsqrt and SP vector versions of fmadd,fnmadd,fmsub,fnmsub)

Pipeline & Instructions

fetch, decode, dispatch, issue, execute, commit

Functional Units



Pipeline & Instructions

fetch, decode, dispatch, issue, **execute**, commit

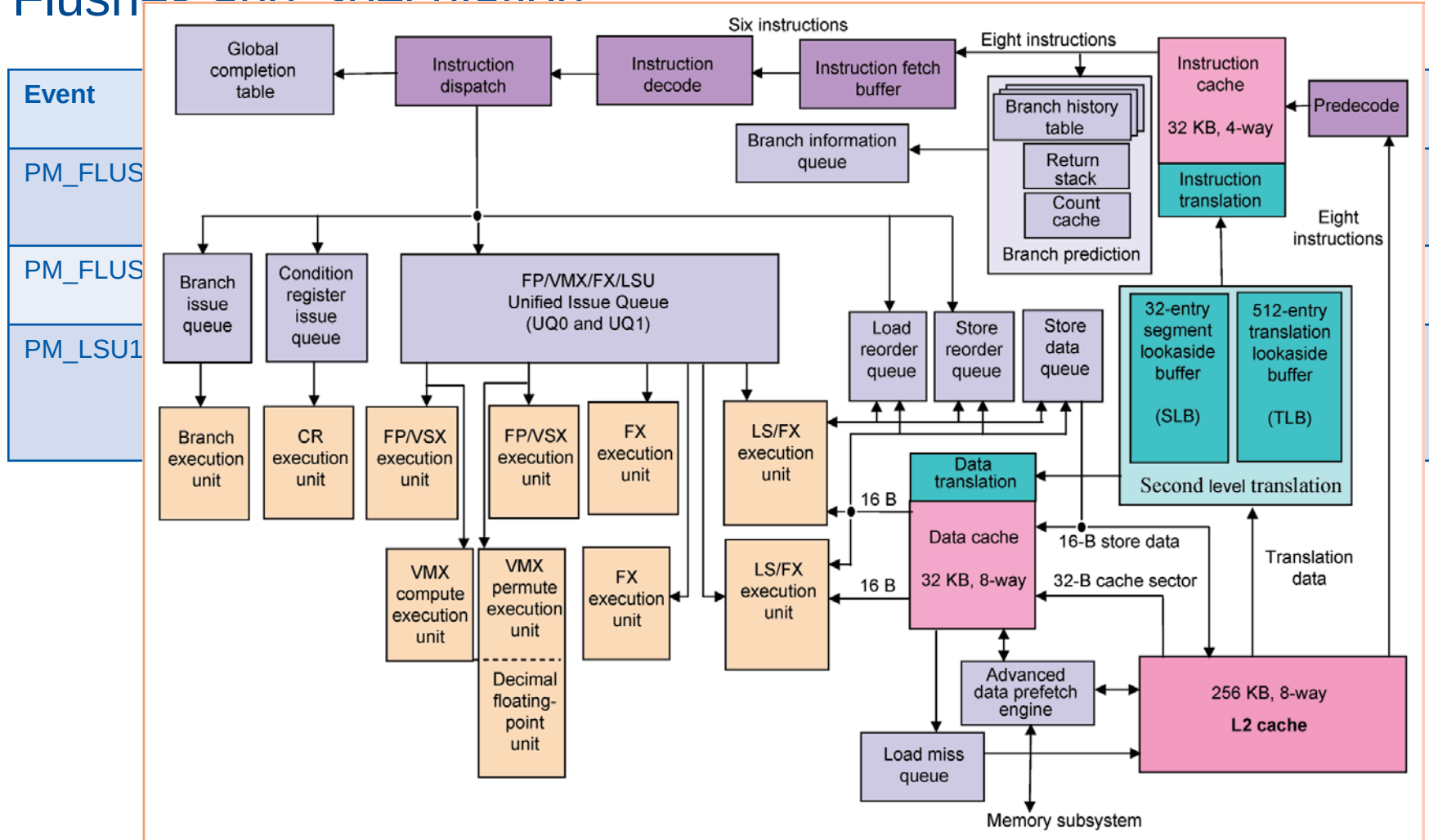
« Flushes and Speculation

Event	Description
PM_FLUSH	Flush (any type). A flush has occurred, this could includes all types of flushes including branch mispredicts, load store unit flushes,partial flushes,completion flushes
PM_FLUSH_BR_MPRED	Flush caused by branch mispredict. A flush was caused by a branch mispredict
PM_LSU1_FLUSH_LRQ LS1	A load issued on LSU1 was flushed because a younger load executed before an older store executed and they had overlapping data OR two loads executed out of order and they have byte overlap and there was a snoop in between to an overlapped byte.

Pipeline & Instructions

fetch, decode, dispatch, issue, execute, commit

Flushes and Speculation



Pipeline & Instructions

fetch, decode, dispatch, issue, execute, **commit**

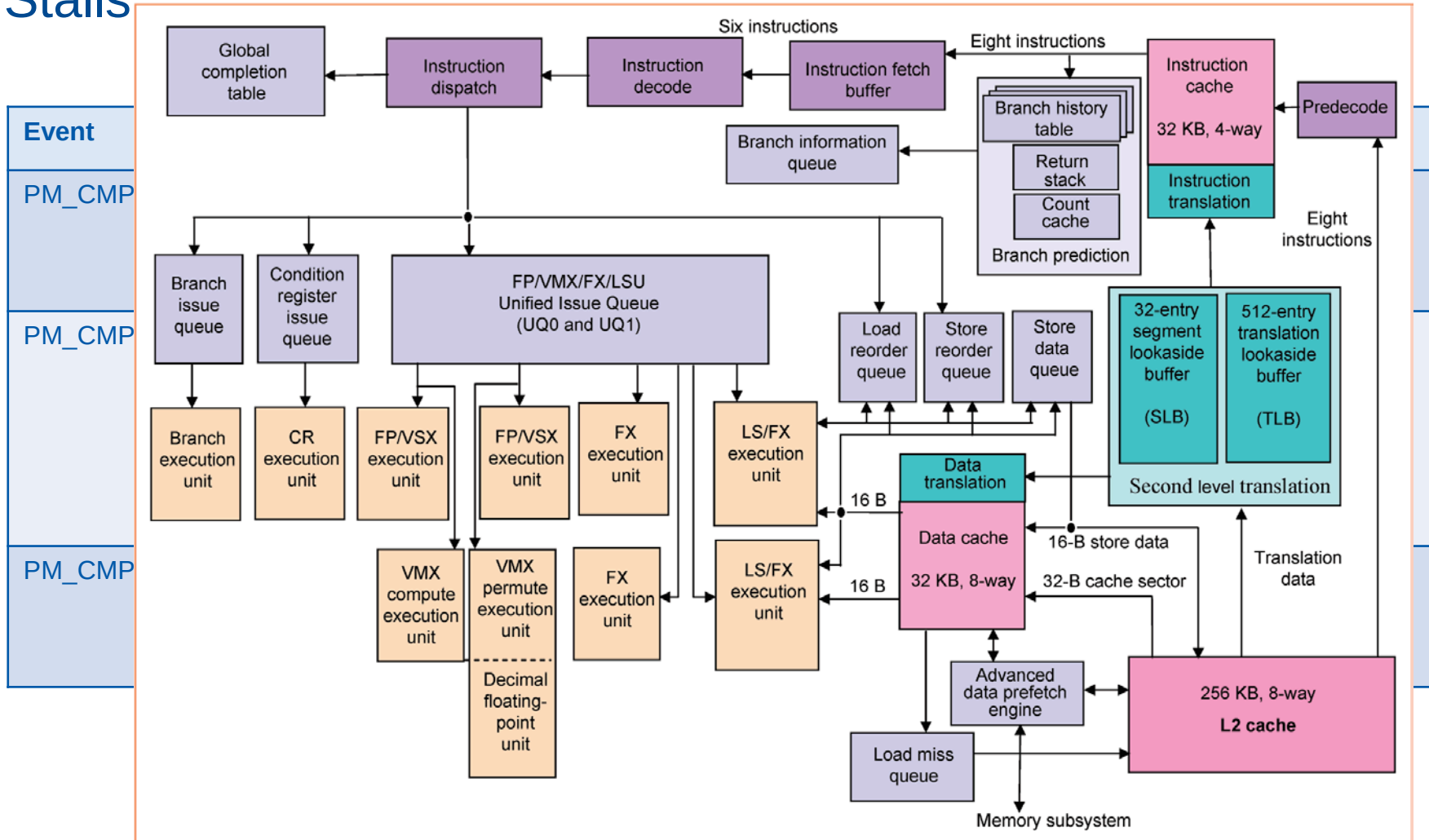
« Stalls

Event	Description
PM_CMPLU_STALL_FXU	Completion stall caused by FXU instruction. Following a completion stall (any period when no groups completed and the group completion table was not empty for that thread) the last instruction to finish before completion resumes was from the Fixed Point Unit.
PM_CMPLU_STALL_DCACHE_MISS	Completion stall caused by D cache miss. Following a completion stall (any period when no groups completed and the group completion table was not empty for that thread) the last instruction to finish before completion resumes suffered a Data Cache Miss. A Data Cache Miss has higher priority than any other Load/Store delay, so if an instruction encounters multiple delays, only the Data Cache Miss will be reported and the entire delay period will be charged to the Data Cache Miss.
PM_CMPLU_STALL_BRU	Completion stall due to BRU. Following a completion stall (any period when no groups completed, while group completion table was not empty for that thread) the last instruction to finish before completion resumes was from the Branch Unit.

Pipeline & Instructions

fetch, decode, dispatch, issue, execute, **commit**

Stalls



Tool: pmctrace

« Tool: pmctrace

- Output trace format: .cvs
- Specify which counters
- Attach to PID
- Frequency of sampling
- Window time to sample

Outline

- « Introduction (10 min.)
- « Data Gathering (25 min.)
- « **Analysis** (45 min.)
- « Modeling (45 min.)
- « Case of Use (45 min.)

DATA ANALYSES

- « Data Formats
- « Data Processing Stages
- « Operational Model
- « Built-In Transformations
- « Plot Support

Data Formats

« INPUT: .csv

« COMPUTE: .potradata

« OUTPUT: .csv, “.plot”

Data Formats: .csv

« .csv format

□ □ □

[illegible]

Data Formats: .potradata

« potradata format

– [] x [] x ... x [] x (dim1, dim2, dim3, ...)



PARAMETERS



ACTUAL VALUES

– In the previous example ...

- Parameters: FREQUENCY, BENCHMARK and CORES
- Values: the counter values

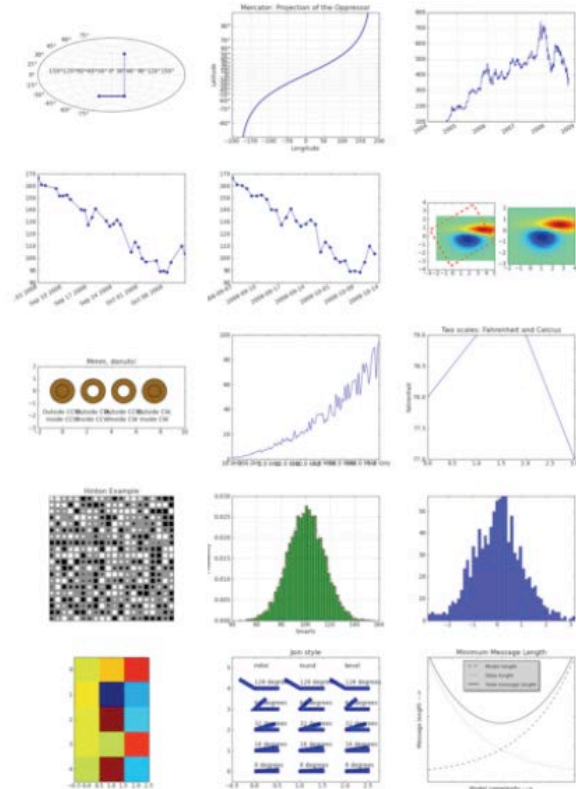
[FREQUENCY] x [BENCHMARK] x [CORES] x [(counter values)]

Data Formats: .plot

« matplotlib

- Exports different formats
 - emf, eps, pdf, png, ps, raw, rgba, svg, svgz, ...

<http://matplotlib.sourceforge.net>

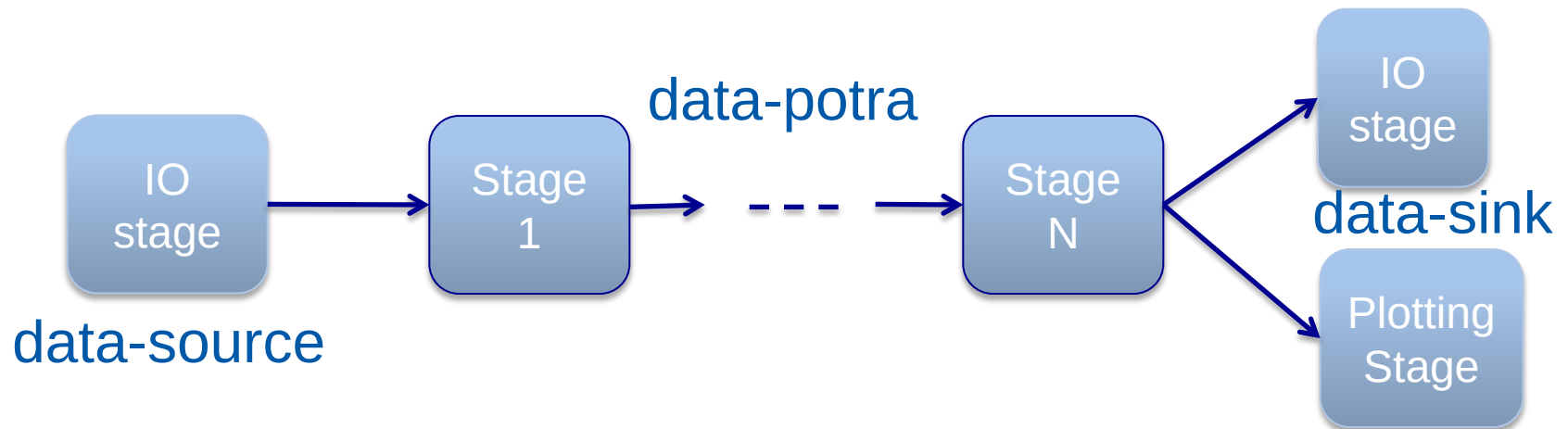


PoTrA Operational Model

⌘ Data transformation: stage

⌘ Along a sequence of stages

- **data-source**: first stage in a sequence of transformations. Input data always in a .csv format
- **data-potra**: middle stage that makes a transformation on the input data. Input and output are both .potradata files.
- **data-sink**: final stage that usually makes no transformations, just emits the output in a particular format (.csv or , .plot)



Built-in Stages

IO

Reader
Csv

Reader
Potra

Writer
Csv

Writer
Potra

Processing

Reshape

Append
Value

Acum
Value

Smooth

Filter

Select

Displace

Slope

FixValue

FixSpikes

Normalize

Flatten

Resample

Reduce

FirstPivot
Cluster

Count
Phases

Compare
Phases

Plotting

LinePlot

PlotDim

PlotDim
3D

Built-in Stages

IO

Reader
Csv

Reader
Potra

Writer
Csv

Writer
Potra

Processing

Reshape

Append
Value

Acum
Value

Smooth

Filter

Select

Displace

Slope

FixValue

FixSpikes

Normalize

Flatten

Resample

Reduce

FirstPivot
Cluster

Count
Phases

Compare
Phases

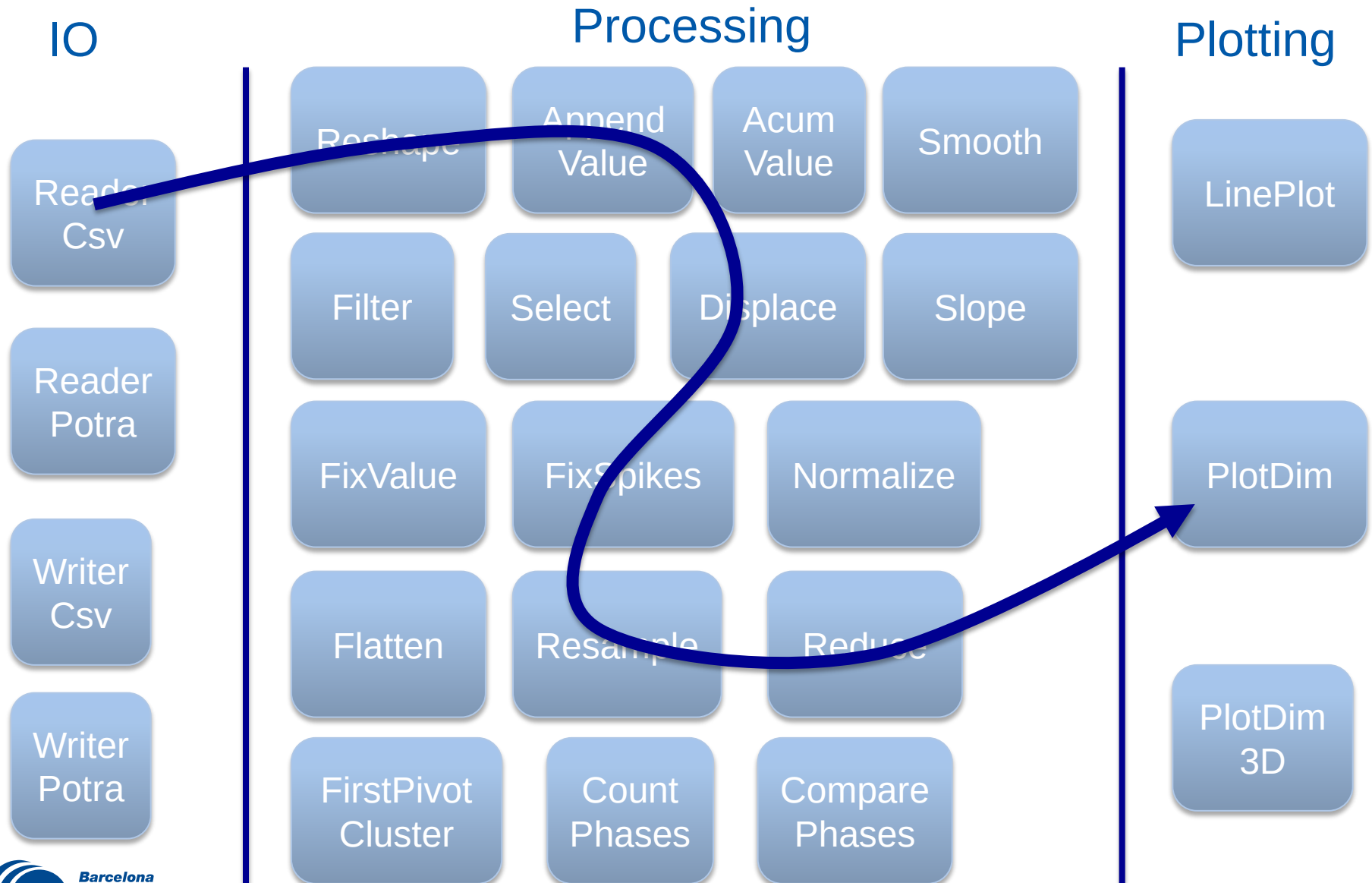
Plotting

LinePlot

PlotDim

PlotDim
3D

Built-in Stages



Built-in Stages

IO

Reader
Csv

Reader
Potra

Writer
Csv

Writer
Potra

Processing

Reshape

Append
Value

Acum
Value

Smooth

Filter

Select

Displace

Slope

FixValue

FixSpikes

Normalize

Flatten

Resample

Reduce

FirstPivot
Cluster

Count
Phases

Compare
Phases

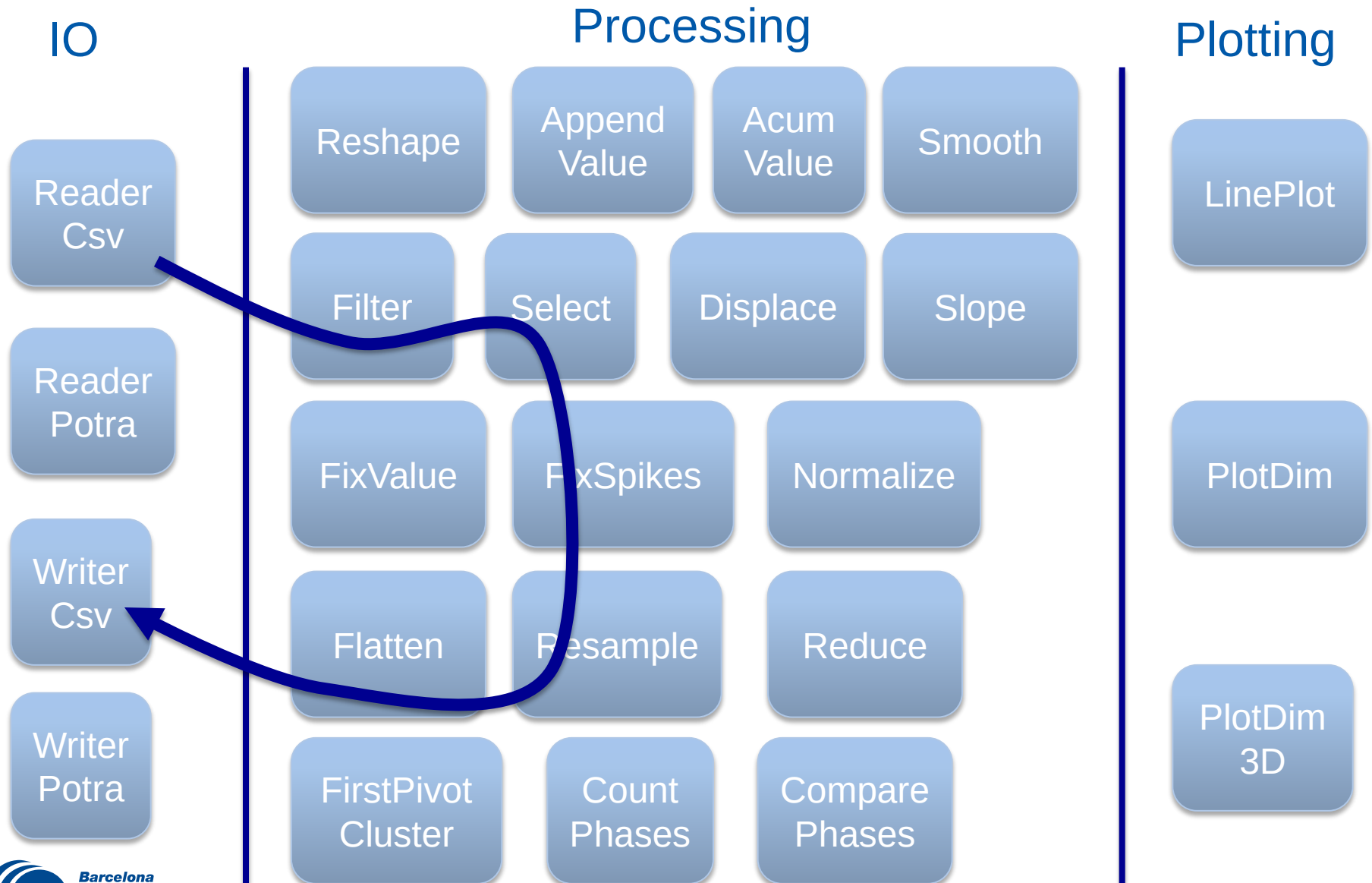
Plotting

LinePlot

PlotDim

PlotDim
3D

Built-in Stages



Built-in Stages

IO

Reader
Csv

Reader
Potra

Writer
Csv

Writer
Potra

Processing

Reshape

Append
Value

Acum
Value

Smooth

Filter

Select

Displace

Slope

FixValue

FixSpikes

Normalize

Flatten

Resample

Reduce

FirstPivot
Cluster

Count
Phases

Compare
Phases

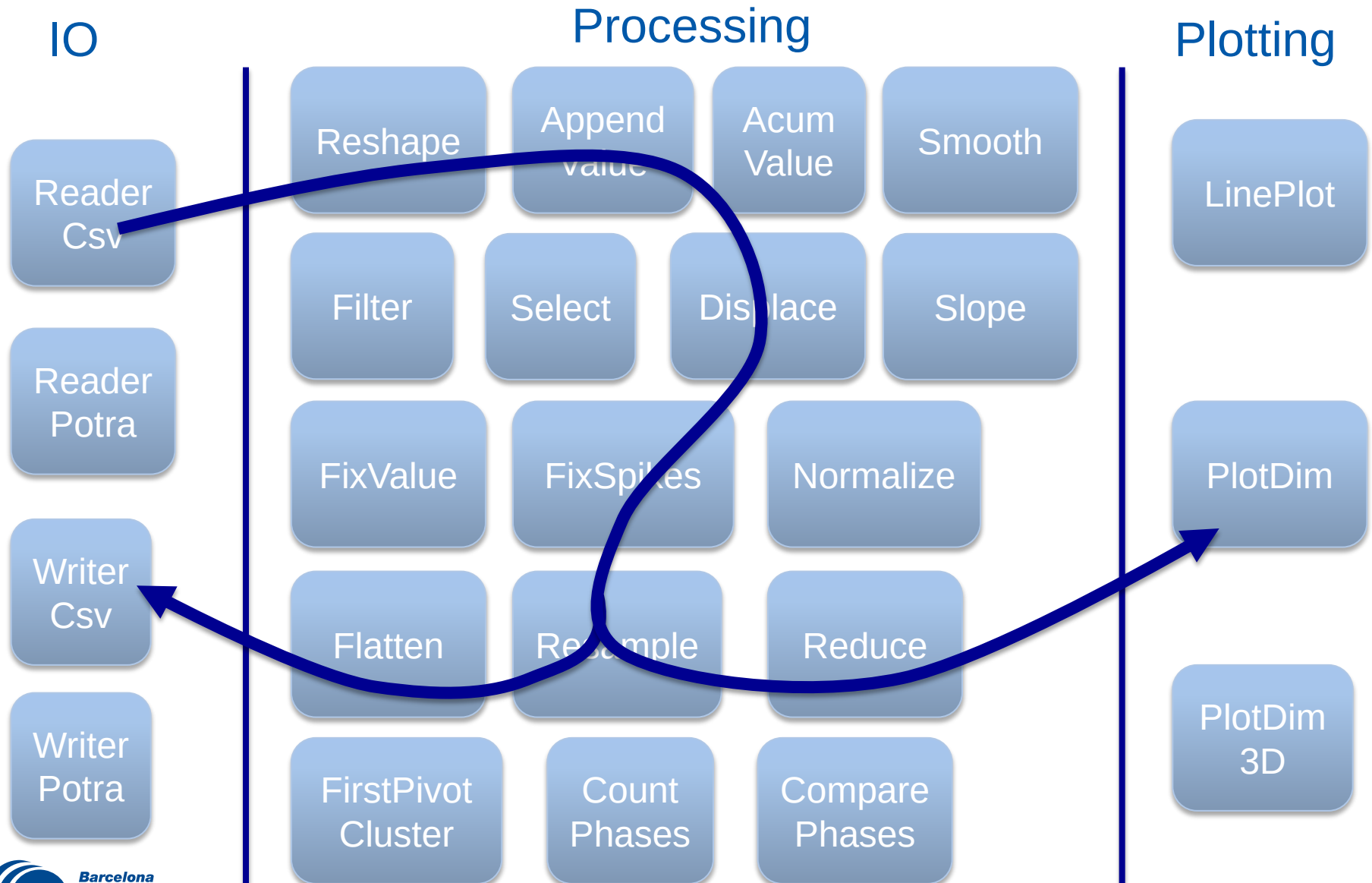
Plotting

LinePlot

PlotDim

PlotDim
3D

Built-in Stages



001-READERCSV-WRITECSV

run.sh

```
#!/bin/bash
set +e
set -x
potra-cmd -c ./001-readerCsv-writerCsv.cfg $@
```

001-readerCsv-writerCsv.cfg

[Input Options]

input = ./input/ ; Input file or directory

inputplugin = ReaderCsv ; Input plugin name. Valid values: ReaderPotra, ReaderCsv

[Output Options]

output = ./output/output.csv ; Output file or directory

outputplugin = WriterCsv ; Output plugin name. Valid values: WriterPotra, WriterCsv

[ReaderCsv]

csv-gather-regexp = @CONFIG@GHZ@BENCHMARK@.csv ; Regular expression to gather the file list and variables.

csv-promote-var = SAMPLE ; Record variable to promote as a dimension variable.

[CONFIG] x [GHZ] x [BENCHMARK] x (sample)

002-readerCsv-writerPotra.cfg

[Input Options]

input = ./input/ ; Input file or directory

inputplugin = ReaderCsv ; Input plugin name. Valid values: ReaderPotra, ReaderCsv

[Output Options]

output = ./output/output.potra ; Output file or directory

outputplugin = WriterPotra ; Output plugin name. Valid values: WriterPotra, WriterCsv

[ReaderCsv]

csv-gather-regexp = @CONFIG@GHZ/@BENCHMARK@.csv ; Regular expression to gather the file list and variables.

csv-promote-var = SAMPLE ; Record variable to promote as a dimension variable.

[WriterPotra]

compression-level = 0 ; Compression level [0:9]

compression-type = None ; Compression type. Valid values: gzip, lzf, szip, None

003-readerPotra-writerPotra.cfg

[Input Options]

input = ./input/output.potra ; Input file or directory

inputplugin = ReaderPotra ; Input plugin name. Valid values: ReaderPotra, ReaderCsv

[Output Options]

output = ./output/output.potra ; Output file or directory

outputplugin = WriterPotra ; Output plugin name. Valid values: WriterPotra, WriterCsv

[WriterPotra]

compression-level = 9 ; Compression level [0:9]

compression-type = gzip ; Compression type. Valid values: gzip, lzf, szip, None

004-readerPotra-writerCsv.cfg

[Input Options]

input = ./input/output.potra ; Input file or directory

inputplugin = ReaderPotra ; Input plugin name. Valid values: ReaderPotra, ReaderCsv

[Output Options]

output = ./output/output.csv ; Output file or directory

outputplugin = WriterCsv ; Output plugin name. Valid values: WriterPotra, WriterCsv

[Input Options]

input = ./input/ ; Input file or directory

inputplugin = ReaderCsv ; Input plugin name. Valid values: ReaderPotra, ReaderCsv

[Output Options]

output = ./output/output.csv ; Output file or directory

outputplugin = WriterCsv ; Output plugin name. Valid values: WriterPotra, WriterCsv

[Processing Options]

plugin = Reshape

[ReaderCsv]

csv-gather-regexp = @CONFIG@GHZ/@BENCHMARK@.csv ; Regular expresion to gather the file list and variables.

csv-promote-var = SAMPLE ; Record variable to promote as a dimension variable.

[Data Reshape]

reshape-dimension = CONFIG ; List of variables for the dimensions.

reshape-dimension = BENCHMARK ; List of variables for the dimensions.

reshape-dimension = SAMPLE ; List of variables for the dimensions.

[Input Options]

input = ./input/ ; Input file or directory

inputplugin = ReaderCsv ; Input plugin name. Valid values: ReaderPotra, ReaderCsv

[Output Options]

output = ./output/output.csv ; Output file or directory

outputplugin = WriterCsv ; Output plugin name. Valid values: WriterPotra, WriterCsv

[Processing Options]

plugin = Reshape

plugin = AppendValue

[ReaderCsv]

csv-gather-regexp = @CONFIG@GHZ/@BENCHMARK@.csv ; Regular expression to gather the file list and variables.

csv-promote-var = SAMPLE ; Record variable to promote as a dimension variable.

[Data Reshape]

reshape-dimension = CONFIG ; List of variables for the dimensions.

reshape-dimension = BENCHMARK ; List of variables for the dimensions.

reshape-dimension = SAMPLE ; List of variables for the dimensions.

[Append Value]

append-value = POWER=((POWER_uW*(-1))-15765.377)

append-value-file = ./Values2append.txt

[Input Options]

input = ./input/ ; Input file or directory

inputplugin = ReaderCsv ; Input plugin name. Valid values: ReaderPotra, ReaderCsv

[Output Options]

output = ./output/output.csv ; Output file or directory

outputplugin = WriterCsv ; Output plugin name. Valid values: WriterPotra, WriterCsv

[Processing Options]

plugin = Reshape

plugin = Filter

[ReaderCsv]

csv-gather-regexp = output.csv ; Regular expresion to gather the file list and variables.

csv-promote-var = SAMPLE ; Record variable to promote as a dimension variable.

csv-promote-var = BENCHMARK ; Record variable to promote as a dimension variable.

csv-promote-var = CONFIG ; Record variable to promote as a dimension variable.

[Data Reshape]

reshape-dimension = CONFIG ; List of variables for the dimensions.

reshape-dimension = BENCHMARK ; List of variables for the dimensions.

reshape-dimension = SAMPLE ; List of variables for the dimensions.

[Filter]

filter-condition = SAMPLE>25 ; Confition to fullfill

filter-condition = SAMPLE<275 ; Confition to fullfill

...

[Processing Options]

plugin = Reshape

plugin = Select

[ReaderCsv]

csv-gather-regexp = output.csv ; Regular expresion to gather the file list and variables.

csv-promote-var = SAMPLE ; Record variable to promote as a dimension variable.

csv-promote-var = BENCHMARK ; Record variable to promote as a dimension variable.

csv-promote-var = CONFIG ; Record variable to promote as a dimension variable.

[Data Reshape]

reshape-dimension = CONFIG ; List of variables for the dimensions.

reshape-dimension = BENCHMARK ; List of variables for the dimensions.

reshape-dimension = SAMPLE ; List of variables for the dimensions.

[Select]

select-value = POWER

select-value = FE

select-value = INT

select-value = FP

select-value = L1

select-value = L2

select-value = MEM

select-value = BPU

select-value = UOPS_RETIREDAANY

009-ACUMVALUE

AccumValue.cfg

...

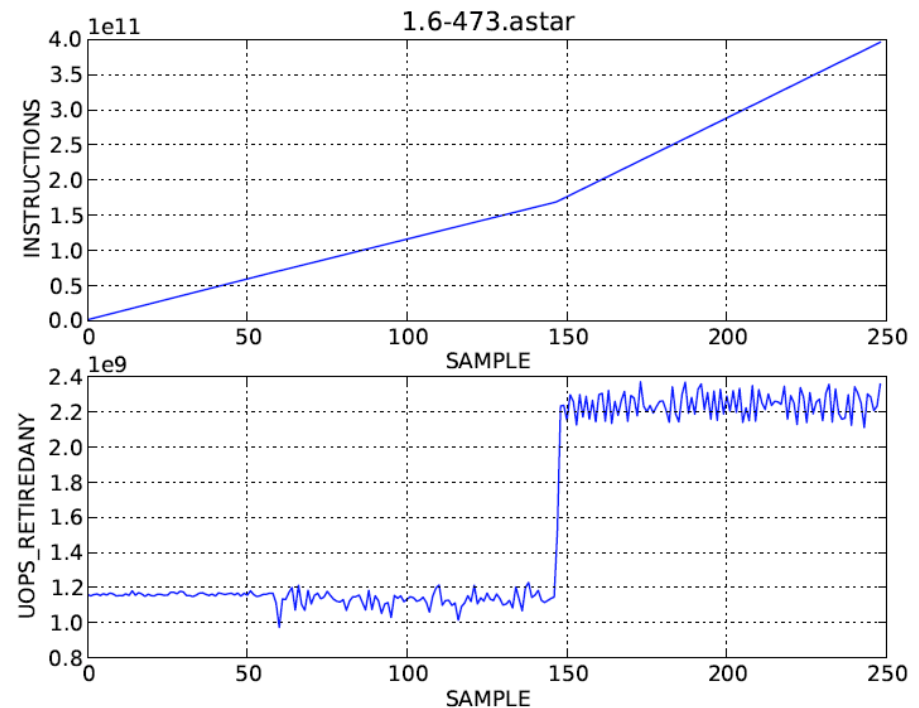
```
[Processing Options]
plugin = Reshape
plugin = AccumValue
plugin = LinePlot
```

```
[ReaderCsv]
csv-gather-regexp = output.csv ; Regular expression to gather the file list and variables.
csv-promote-var = SAMPLE ; Record variable to promote as a dimension variable.
csv-promote-var = BENCHMARK ; Record variable to promote as a dimension variable.
csv-promote-var = CONFIG ; Record variable to promote as a dimension variable.
```

```
[Data Reshape]
reshape-dimension = CONFIG ; List of variables for the dimensions.
reshape-dimension = BENCHMARK ; List of variables for the dimensions.
reshape-dimension = SAMPLE ; List of variables for the dimensions.
```

```
[AccumValue]
accum-value = INSTRUCTIONS=UOPS_RETIREDANY
```

```
[Common Plot Options]
plot-valuenam = INSTRUCTIONS
plot-valuenam = UOPS_RETIREDANY
plot-split = True
plot-outdir = ./plots/ALL
plot-format = pdf
plot-join-cmd = pdjoin --outfile ./plots/output.pdf @FILES@ ;
```



010-DISPLACE

Displace.cfg

[Append Value]

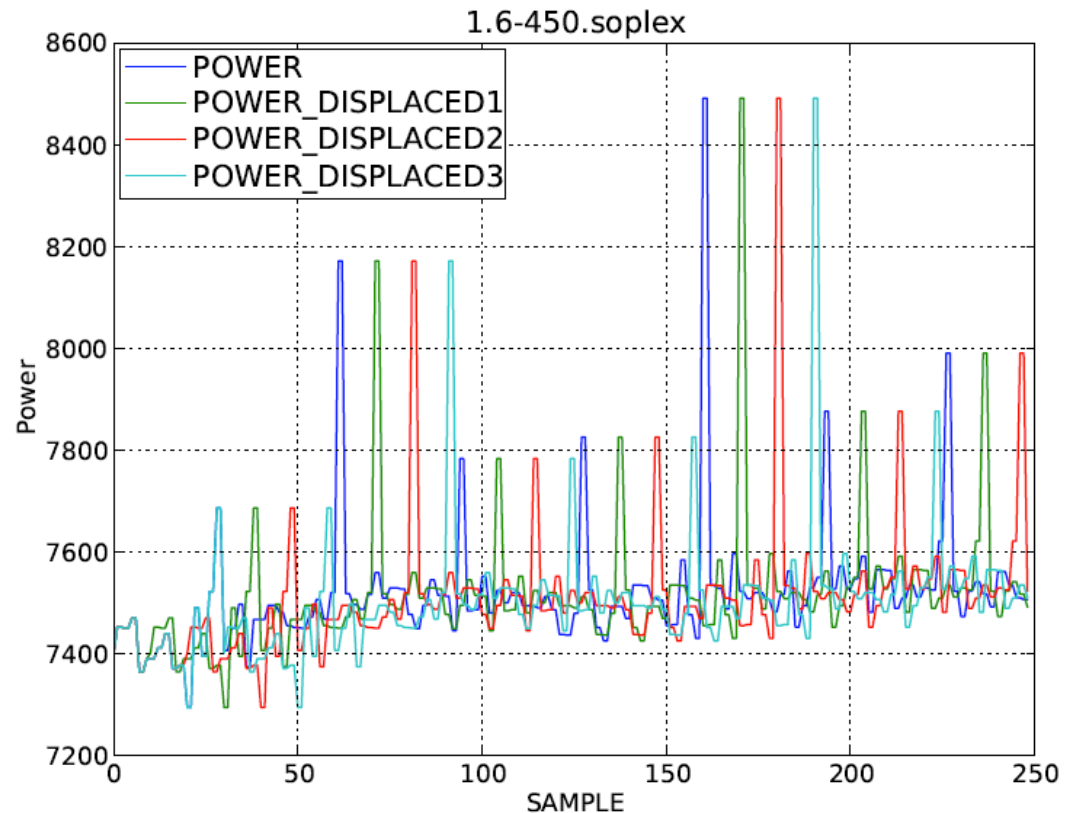
```
append-value = POWER_DISPLACED1=POWER  
append-value = POWER_DISPLACED2=POWER  
append-value = POWER_DISPLACED3=POWER
```

[Displace]

```
displace-valuenam = POWER_DISPLACED1  
displace-records = 10  
displace-valuenam = POWER_DISPLACED2  
displace-records = 20  
displace-valuenam = POWER_DISPLACED3  
displace-records = 30
```

[Common Plot Options]

```
plot-valuenam = POWER  
plot-valuenam = POWER_DISPLACED1  
plot-valuenam = POWER_DISPLACED2  
plot-valuenam = POWER_DISPLACED3  
plot-split = False  
plot-outdir = ./plots/ALL  
plot-format = pdf  
plot-join-cmd = pdftjoin --outfile ./plots/output.pdf @FILES@ ;  
plot-legendtitle = True  
plot-ylabel = Power
```



011-FIX VALUE

[FixValue]

fix-valuenam = POWER_FIX

fix-valuenam = POWER_FIX

fix-condition = >((2*-std-)+-nanmean-)

fix-condition = <((-2*-std-)+-nanmean-)

fix-type = previous-value

fix-type = previous-value

fix-valuenam = FE_FIX

fix-valuenam = FE_FIX

fix-condition = >((2*-std-)+-nanmean-)

fix-condition = <((-2*-std-)+-nanmean-)

fix-type = previous-value

fix-type = previous-value

[Common Plot Options]

plot-valuenam = POWER

plot-valuenam = POWER_FIX

plot-valuenam = FE

plot-valuenam = FE_FIX

plot-split = True

plot-outdir = ./plots/ALL

plot-format = pdf

plot-join-cmd = pdftjoin --outfile ./plots/output.p

plot-legendtitle = True

plot-ylabel = Power

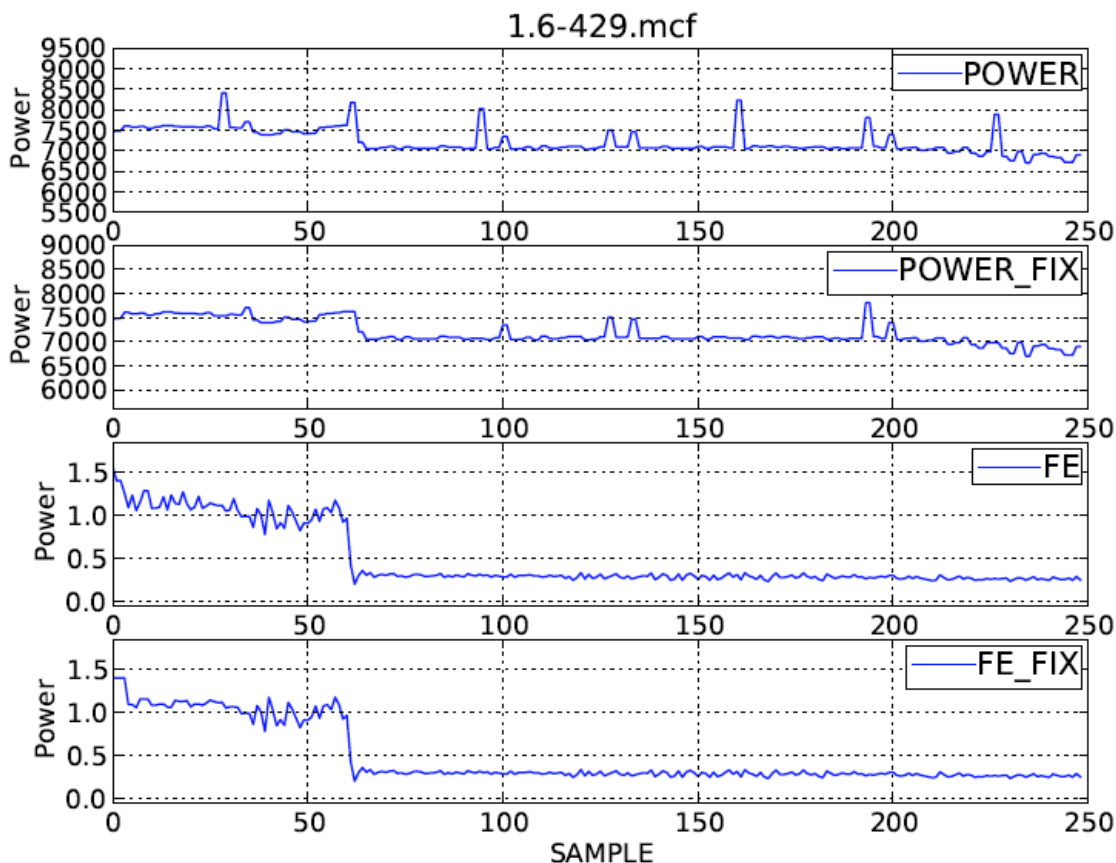
plot-ylim = <1000,>1000

plot-ylim = <1000,>1000

plot-ylim = <0.25,>0.25

plot-ylim = <0.25,>0.25

FixValue.cfg



012-FIX SPIKES

FixSpikes.cfg

[FixSpikes]

fixspikes-valuenname = POWER_FIXSPIKES

fixspikes-base-valuenname = FE

fixspikes-base-valuenname-minvariation = 0.3

fixspikes-valuenname-minvariation = 250

fixspikes-window-size = 6

[Common Plot Options]

plot-valuenname = POWER

plot-valuenname = POWER_FIXSPIKES

plot-split = False

plot-outdir = ./plots/ALL

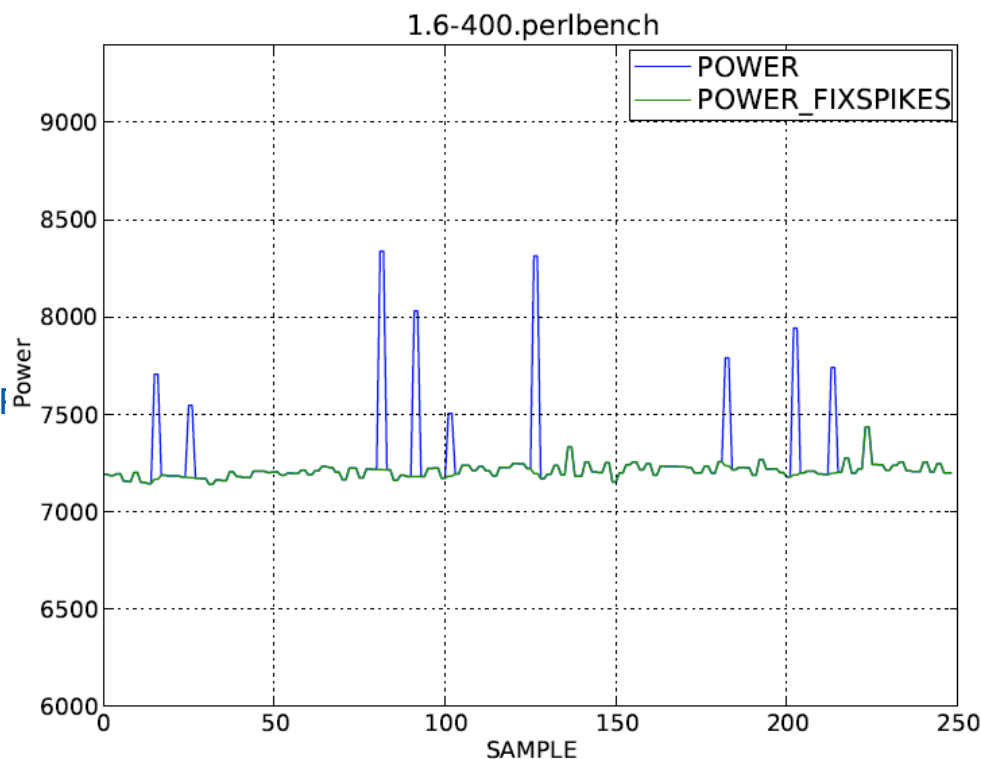
plot-format = pdf

plot-join-cmd = pdftopdf --outfile ./plots/output.pdf @I

plot-legendtitle = True

plot-ylabel = Power

plot-ylim = <1000,>1000



013-RESAMPLE

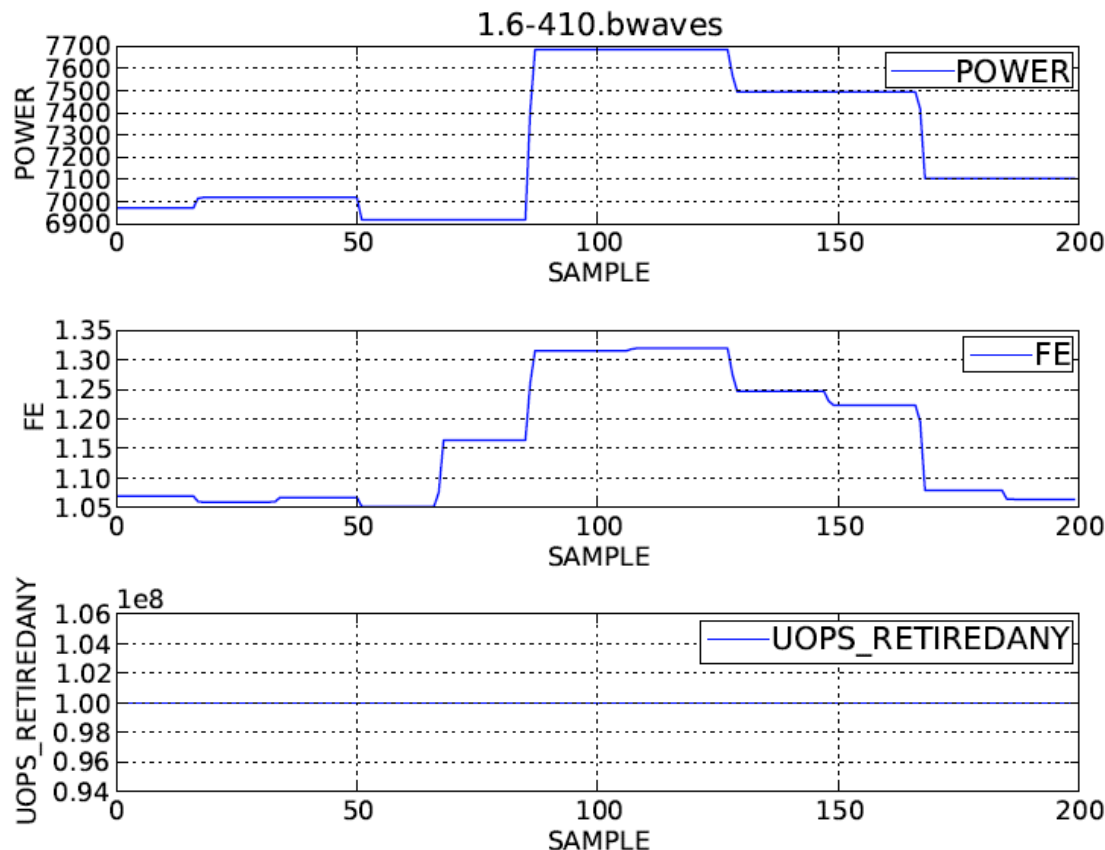
Resample.cfg

[Resample]

```
resample-valuenam = UOPS_RETIREDAANY
resample-interval-events = 100000000
resample-maxevents = 20000000000
resample-keep-values = POWER
resample-keep-values = FE
resample-keep-values = INT
resample-keep-values = FP
resample-keep-values = L1
resample-keep-values = L2
resample-keep-values = MEM
resample-keep-values = BPU
resample-keep-values = POWER_DISPLACED1
resample-keep-values = POWER_DISPLACED2
resample-keep-values = POWER_DISPLACED3
resample-keep-values = POWER_FIXSPIKES
```

[Common Plot Options]

```
plot-valuenam = POWER
plot-valuenam = FE
plot-valuenam = UOPS_RETIREDAANY
plot-split = True
plot-outdir = ./plots/ALL
plot-format = pdf
plot-join-cmd = pdjoin --outfile ./plots/output.pdf
plot-legendtitle = True
plot-vs = 0.6
```



[Smooth]

smooth-level = 3

smooth-keep-valuenam = POWER

smooth-keep-valuenam = POWER_FIXSPIKES

[Common Plot Options]

plot-valuenam = POWER

#plot-valuenam = POWER_SMOOTH

plot-valuenam = POWER_FIXSPIKES

plot-valuenam = POWER_FIXSPIKES_SMC

plot-split = False

plot-outdir = ./plots/ALL

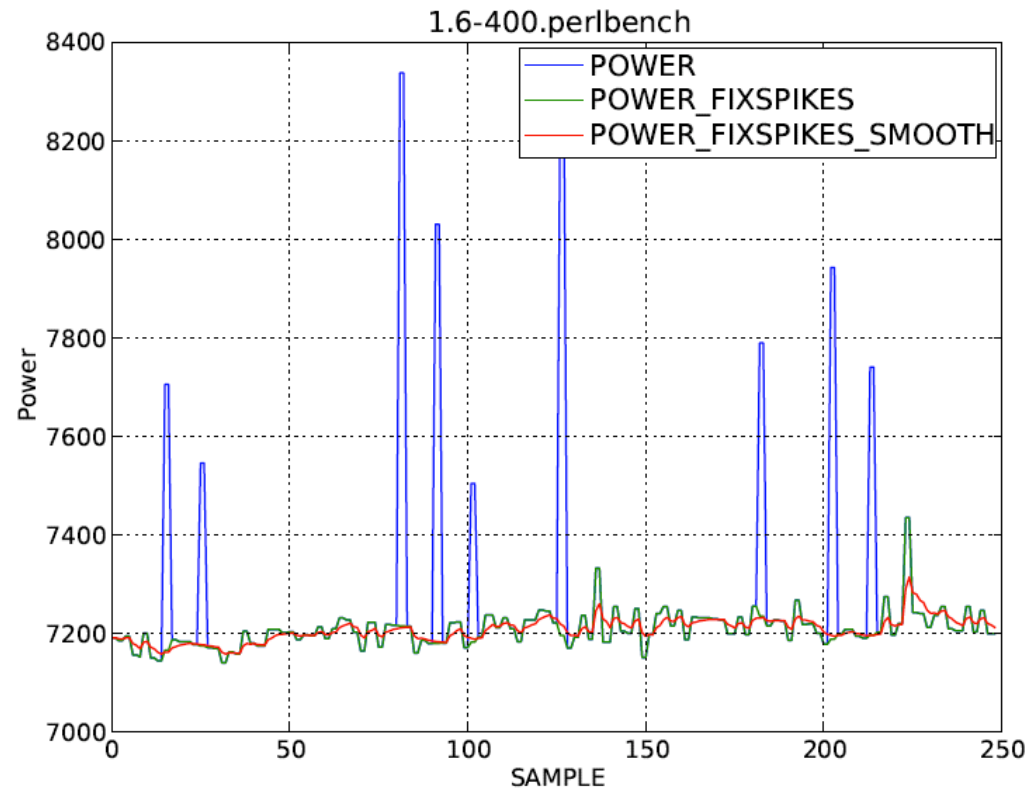
plot-format = pdf

plot-join-cmd = pdftjoin --outfile ./plots/output.p

plot-legendtitle = True

plot-vs = 0.6

plot-ylabel = Power



017-FIRST PIVOT CLUSTER

FirstPivotCluster.cfg

[First Pivot Cluster]

fpc-value = POWER

fpc-newvalue = POWER_FPC

fpc-value = POWER_FIXSPIKES

fpc-newvalue = POWER_FIXSPIKES_FPC

fpc-threshold = 250

fpc-threshold = 250

[Common Plot Options]

plot-size = 6.5x9

plot-fs = 10

plot-valuenam = POWER

plot-valuenam = POWER_FPC_LENGTH

plot-valuenam = POWER_FPC_DIFF

plot-valuenam = POWER_FIXSPIKES

plot-valuenam = POWER_FIXSPIKES_FPC_LENGTH

plot-valuenam = POWER_FIXSPIKES_FPC_DIFF

plot-split = True

plot-outdir = ./plots/ALL

plot-format = pdf

plot-join-cmd = pdffjoin --outfile ./plots/output.pdf @FILES@ ;

plot-legendtitle = True

plot-ylabel = Power

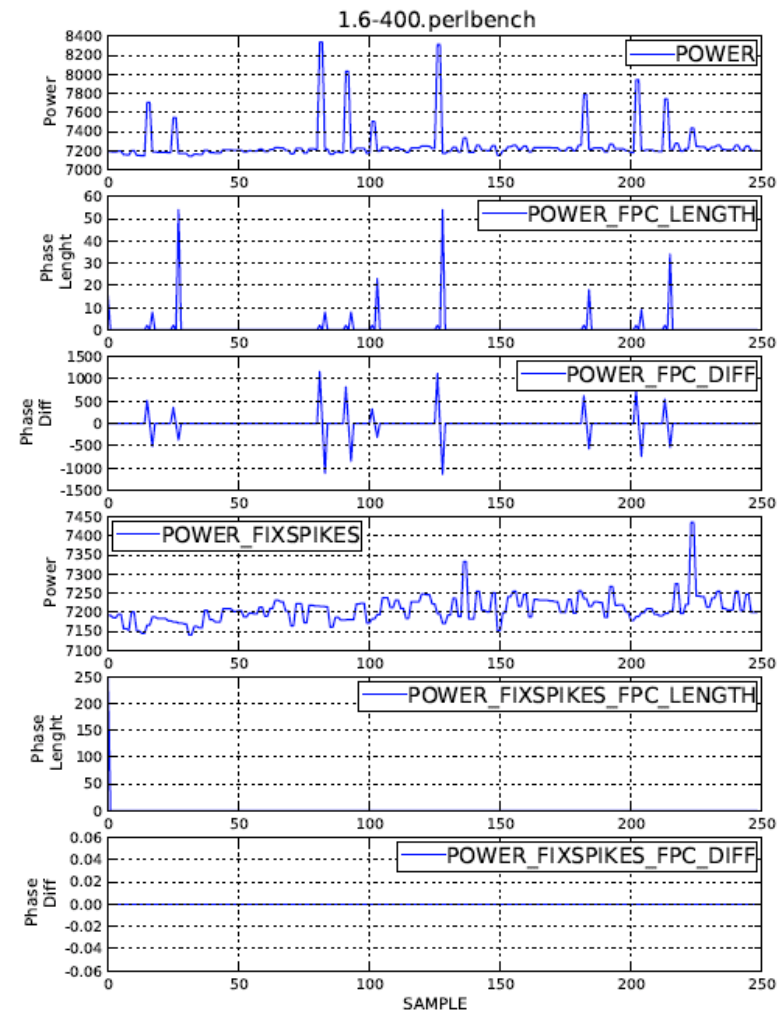
plot-ylabel = Phase\nLenght

plot-ylabel = Phase\nDiff

plot-ylabel = Power

plot-ylabel = Phase\nLenght

plot-ylabel = Phase\nDiff



018-FLATTEN

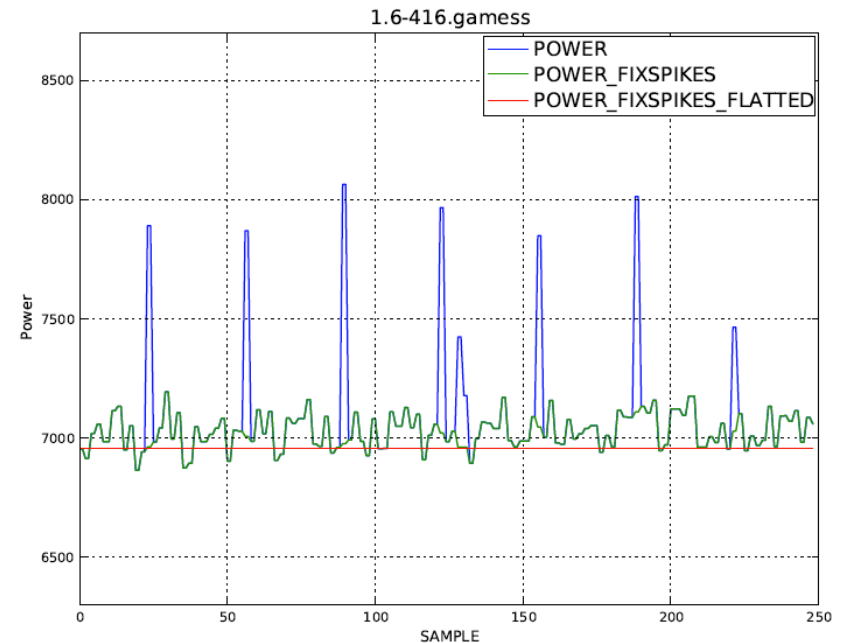
Flatten.cfg

[Flatten]

```
flatten-condition = POWER_FPC_LENGTH>0  
flatten-value = POWER_FLATTED  
flatten-condition = POWER_FIXSPIKES_FPC_LENGTH>0  
flatten-value = POWER_FIXSPIKES_FLATTED
```

[Common Plot Options]

```
plot-fs = 10  
plot-valuenam = POWER  
plot-valuenam = POWER_FIXSPIKES  
#plot-valuenam = POWER_FLATTED  
plot-valuenam = POWER_FIXSPIKES_FLATTED  
plot-split = False  
plot-outdir = ./plots/ALL  
plot-format = pdf  
plot-join-cmd = pdftjoin --outfile ./plots/output.pdf @FILES@ ;  
plot-legendtitle = True  
plot-ylabel = Power  
plot-ylim = <500,>500
```



019-COUNT PHASES

[Count Phases]

```
count-ph-value = POWER_FPC
count-ph-variation-intervals = 0,250,375,500,625,750,875,1000,99999
count-ph-duration-intervals = 0,2,3,4,5,6,7,8,9,10,30,60,9999
count-ph-skip-min-variation = 0
count-ph-skip-min-duration = 0
#count-ph-join = False
```

[PlotDim3D]

```
p3d-type = Surface
p3d-surface-contour = True
```

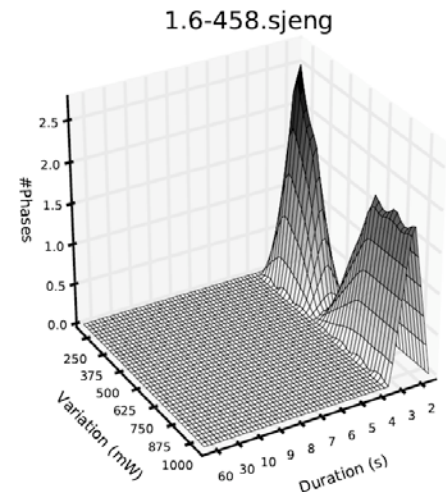
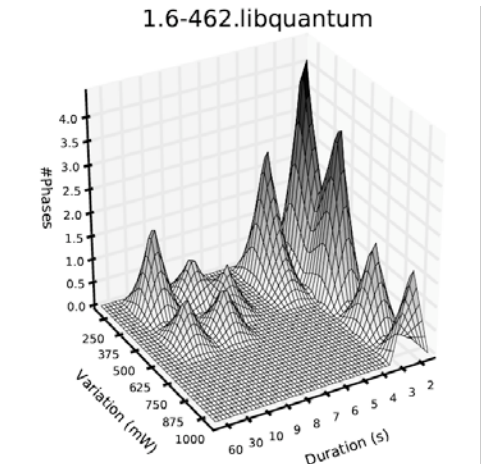
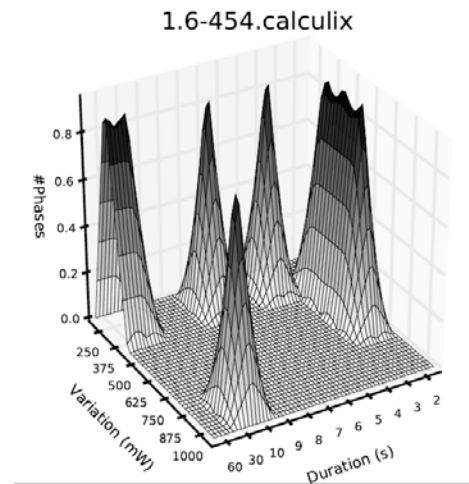
[Common 3D Plot Options]

```
plot3d-elevation=30
plot3d-rotation=60
plot3d-interpolate = 5
plot3d-zlabel = #Phases
```

[Common Plot Options]

```
plot-outdir = ./plots/ALL
plot-size = 2.2x2.2
plot-margins = 0.02x1x0.02x1
plot-split = True
plot-xlabel = \n\nDuration (s)
plot-vs = 0.5
plot-valuenam = PHASES
plot-ylabel = \n\nVariation (mW)
plot-fs = 6
plot-lfs = 6
plot-title = @CONFIG@-@BENCHMARK@
plot-color=k
plot-autoticks=True
plot-legendtitle=# Phases
plot-format=pdf
plot-join-cmd = pdjoin --outfile ./plots/output.pdf @FILES@ ;
```

CountPhases.cfg



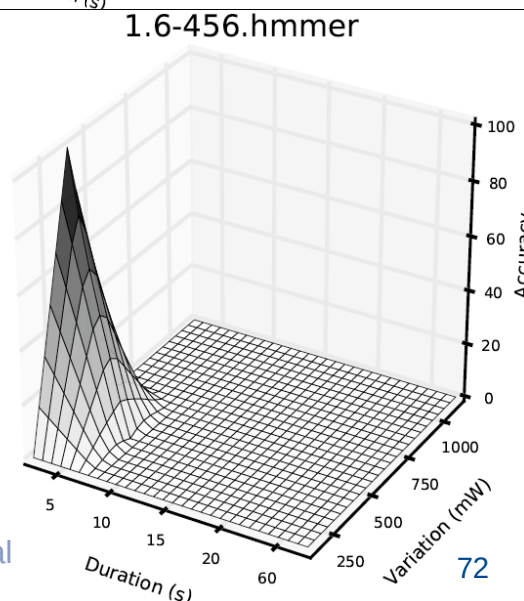
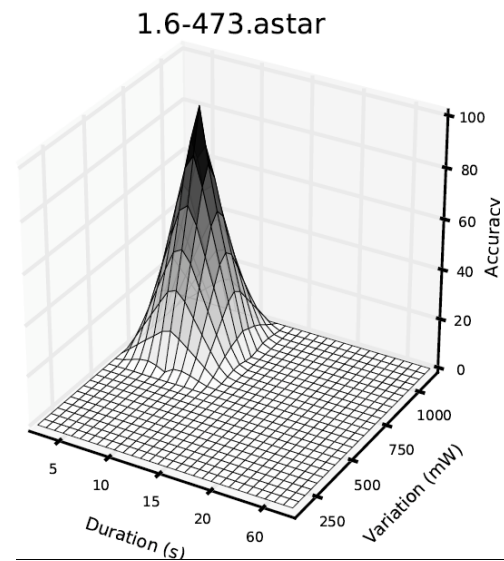
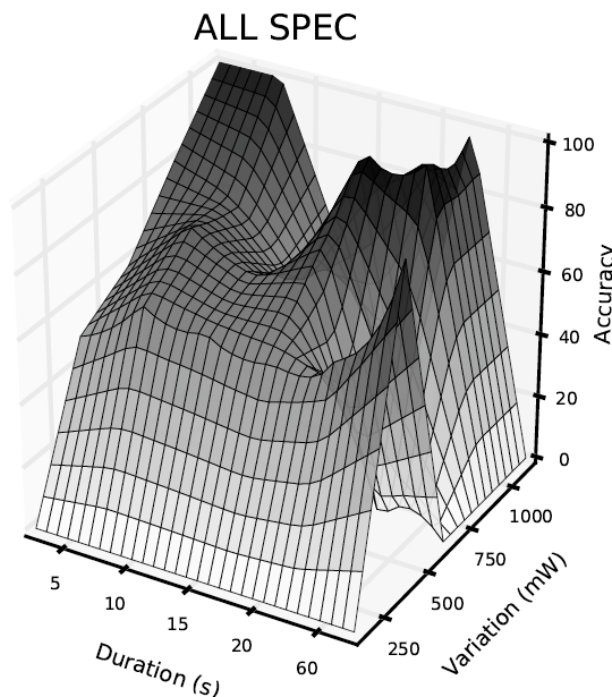
020-COMPARE PHASES

ComparePhases-Accuracy.cfg

```
[Compare Phases]
comp-ph-base-value = POWER_FPC
comp-ph-value = POWER_FIXSPIKES_FPC
comp-ph-variation-intervals = 0,250,500,750,1000,99999
comp-ph-duration-intervals = 2.5,5,10,15,20,60,9999
comp-ph-skip-min-variation = 1500
comp-ph-skip-min-duration = 2
comp-ph-proximity-threshold = 1
#comp-ph-join = True
```

```
[PlotDim3D]
p3d-type = Surface
p3d-surface-contour = True
```

```
[Common 3D Plot Options]
plot3d-elevation=30
plot3d-rotation=-60
plot3d-interpolate = 5
plot3d-zlabel = Accuracy
plot3d-zlim = 0,100
```



022-REDUCE

Reduce.cfg

[Reduce]

default-reduce-function = gmean

reduce-function-map = MINFE=min,MAXFE=max

[LinePlot]

pl-type = Bar

[Common Plot Options]

plot-valuenam = FE

plot-valuenam = MINFE

plot-valuenam = MAXFE

plot-autoticks = True

plot-xticksrotation = 90

plot-margins = 0.1x0.9x0.3x0.9

plot-split = False

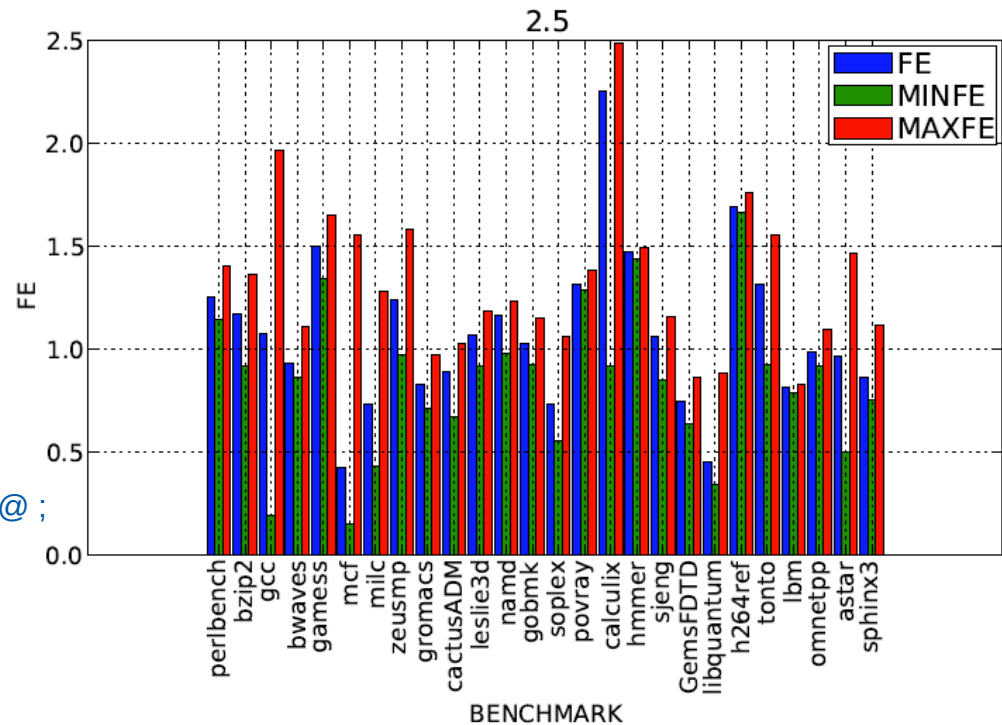
plot-outdir = ./plots/ALL

plot-format = pdf

plot-join-cmd = pdfjoin --outfile ./plots/output.pdf @FILES@ ;

plot-legendtitle = True

plot-ylabel = FE



024-NORMALIZE

Normalize.cfg

[Normalize]

normalize-to-value = 1.6

normalize-dimension = CONFIG

[Filter]

filter-condition = CONFIG>1.6 ; Confiton to fullfill

[PlotDim]

pd-dimensionname = CONFIG

pd-type = Bar

[Common Plot Options]

plot-valuenname = POWER_NORM

plot-autoticks = True

plot-xticksrotation = 90

plot-margins = 0.1x0.9x0.3x0.9

plot-split = False

plot-outdir = ./plots/ALL

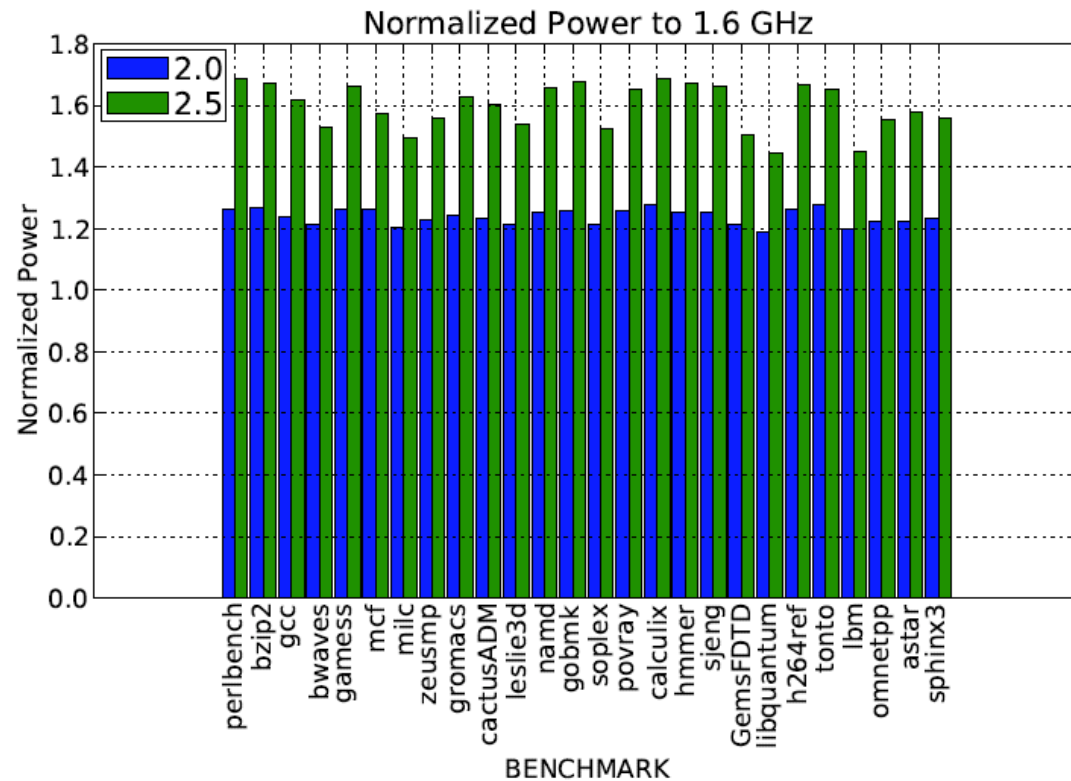
plot-format = pdf

plot-join-cmd = pdjoin --outfile ./plots/output.pdf @FILES@ ;

plot-legendtitle = True

plot-ylabel = Normalized Power

plot-title = Normalized Power to 1.6 GHz



[Slope]

slope-dimension = CONFIG

slope-numeric = True

[PlotDim]

pd-dimensionname = CONFIG

pd-type = Bar

[Common Plot Options]

plot-valuenname = POWER_NORM

plot-autoticks = True

plot-xticksrotation = 90

plot-margins = 0.1x0.9x0.3x0.9

plot-split = False

plot-outdir = ./plots/ALL

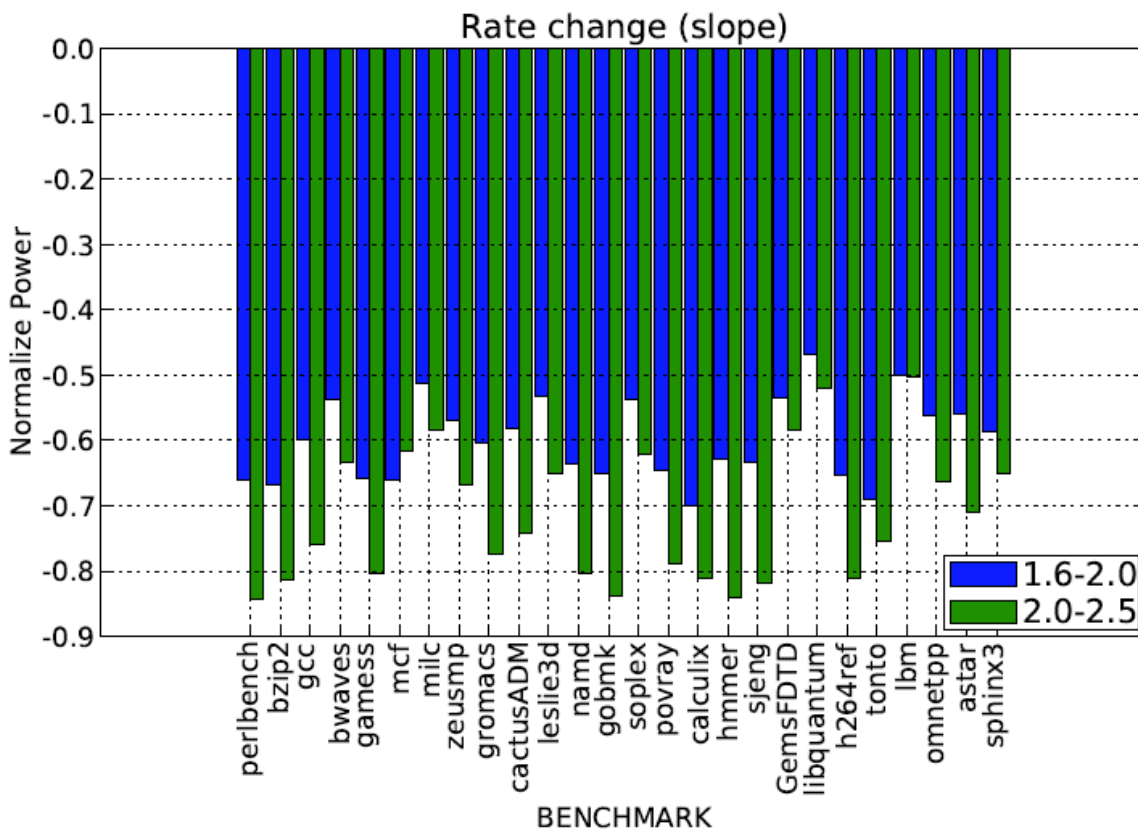
plot-format = pdf

plot-join-cmd = pdfjoin --outfile ./plots/output

plot-legendtitle = True

plot-ylabel = Normalize Power

plot-title = Rate change (slope)





**Barcelona
Supercomputing
Center**

Centro Nacional de Supercomputación

QUESTIONS?